Helsinki University of Technology
Publications in Computer and Information Science

Report E10
March 2007

NATURAL CONJUGATE GRADIENT IN VARIATIONAL INFERENCE

Antti Honkela Matti Tornio Tapani Raiko Juha Karhunen

TEKNILLINEN KORKEAKOULU
TEKNISKA HÖGSKOLAN
HELSINKI UNIVERSITY OF TECHNOLOGY
TECHNISCHE UNIVERSITÄT HELSINKI
UNIVERSITE DE TECHNOLOGIE D'HELSINKI

Distribution:
Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory of Computer and Information Science
P.O. Box 5400
FI-02015 TKK, Finland
Tel. +358-9-451 3267
Fax +358-9-451 3277

This report is downloadable at
http://www.cis.hut.fi/Publications/

ISBN 978-951-22-8701-7
ISSN 1796-2803

Natural Conjugate Gradient in Variational Inference

Antti Honkela **Matti Tornio** **Tapani Raiko** **Juha Karhunen**
Adaptive Informatics Research Centre, Helsinki University of Technology
P.O. Box 5400, FI-02015 TKK, Finland
{Antti.Honkela, Matti.Tornio, Tapani.Raiko, Juha.Karhunen}@tkk.fi
http://www.cis.hut.fi/projects/bayes/

Abstract

Variational methods for approximate inference in machine learning often adapt a parametric probability distribution to optimize a given objective function. This view is especially useful when applying variational Bayes (VB) to models outside the conjugate-exponential family. For them, variational EM algorithms are not easily available, and gradient-based methods are often used as alternatives. However, regular gradient methods ignore the Riemannian geometry of the manifold of probability distributions, thus leading to slow convergence. We propose using the Riemannian structure of the approximations and the natural gradient to speed up a conjugate gradient method for variational learning and inference. As the form of the approximating distribution is often very simple, the natural gradient can be used for both model parameters and latent variables without significant computational overhead. Experiments in variational Bayesian learning of nonlinear state-space models for real speech data show more than ten-fold speedups over alternative learning algorithms.

1 INTRODUCTION

Variational Bayesian (VB) methods provide an efficient and often sufficiently accurate deterministic approximation to exact Bayesian learning. Most work on variational methods has focused on the class of conjugate exponential models for which simple EM-like learning algorithms can be derived easily (Ghahramani and Beal, 2001; Winn and Bishop, 2005).

Nevertheless, there are many interesting more complicated models which are not in the conjugate exponential family. Similar variational approximations have been applied for many such models (Barber and Bishop, 1998; Seeger, 2000; Lappalainen and Honkela, 2000; Valpola and Karhunen, 2002; Valpola et al., 2004; Honkela and Valpola, 2005). The approximating distribution $q(\boldsymbol{\theta}|\boldsymbol{\xi})$, where $\boldsymbol{\theta}$ includes both model parameters and latent variables, is often restricted to be Gaussian with a somehow restricted covariance. Values of the variational parameters $\boldsymbol{\xi}$ can be found by using a gradient-based optimization algorithm.

When applying a generic optimization algorithm for such problem, a lot of background information on the geometry of the problem is lost. The parameters $\boldsymbol{\xi}$ of $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ often have different roles, as the distribution has separate location, shape, and scale parameters. This implies that the geometry of the problem is in most, especially more complicated cases, not Euclidean.

Information geometry studies the Riemannian geometric structure of the manifold of probability distributions (Amari, 1985). It has previously been applied to derive efficient natural gradient learning rules for maximum likelihood algorithms to problems such as independent component analysis (ICA) (Yang and Amari, 1997; Amari, 1998) and multilayer perceptron (MLP) networks (Amari, 1998) as well as to analyze the properties of general EM (Amari, 1995), mean-field variational learning (Tanaka, 2001), and online VB EM (Sato, 2001).

In this paper we propose using the Riemannian structure of the distributions $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ to derive more efficient algorithms for approximate inference and especially mean field type VB. The method can be used to jointly optimize all the parameters $\boldsymbol{\xi}$ of the approximation $q(\boldsymbol{\theta}|\boldsymbol{\xi})$, or in conjunction with variational EM for some parameters. The method is especially useful for models that are not in the conjugate exponential family, such as nonlinear models (Barber and Bishop, 1998; Seeger, 2000; Lappalainen and Honkela, 2000; Valpola and Karhunen, 2002; Honkela and Valpola, 2005) or non-conjugate variance models (Valpola et al., 2004) that may not have a tractable exact variational EM algorithm.

2 INFORMATION GEOMETRY AND NATURAL GRADIENT

Let $\mathcal{F}(\boldsymbol{\xi})$ be a scalar function defined on the manifold $S = \{\boldsymbol{\xi} \in \mathbf{R}^n\}$. If S is a Euclidean space and the coordinate system $\boldsymbol{\xi}$ is orthonormal, the length of a small incremental vector $\mathbf{w}$ is given by

$$|\mathbf{w}|^2 = \sum_{i=1}^{n} w_i^2, \tag{1}$$

where w_i is the ith component of the vector $\mathbf{w}$. The direction of steepest ascent, i.e. the direction that maximizes $\mathcal{F}(\boldsymbol{\xi} + \mathbf{w})$ under the constraint $|\mathbf{w}|^2 = \epsilon^2$ for a sufficiently small constant ϵ, is given by the gradient $\nabla\mathcal{F}(\boldsymbol{\xi})$.

If the space S is a curved manifold, there is no orthonormal coordinate system and the the length of a vector $\mathbf{w}$ differs from the value given by Eq. (1). Riemannian manifolds are an important class of curved manifolds, where the length is given by the positive quadratic form

$$|\mathbf{w}|^2 = \sum_{i,j} g_{ij}(\boldsymbol{\xi}) w_i w_j. \tag{2}$$

The $n \times n$ matrix $\mathbf{G}(\boldsymbol{\xi}) = (g_{ij}(\boldsymbol{\xi}))$ is called the Riemannian metric tensor and it may depend on the point of origin $\boldsymbol{\xi}$. On a Riemannian manifold, the direction of steepest ascent is given by the natural gradient (Amari, 1998)

$$\tilde{\nabla}\mathcal{F}(\boldsymbol{\xi}) = \mathbf{G}^{-1}(\boldsymbol{\xi})\nabla\mathcal{F}(\boldsymbol{\xi}). \tag{3}$$

For the space of probability distributions $q(\boldsymbol{\theta}|\boldsymbol{\xi})$, the most common Riemannian metric tensor is given by the Fisher information (Amari, 1985)

$$\begin{aligned} I_{ij}(\boldsymbol{\xi}) = g_{ij}(\boldsymbol{\xi}) &= E\left\{\frac{\partial \ln q(\boldsymbol{\theta}|\boldsymbol{\xi})}{\partial \xi_i}\frac{\partial \ln q(\boldsymbol{\theta}|\boldsymbol{\xi})}{\partial \xi_j}\right\} \\ &= E\left\{-\frac{\partial^2 \ln q(\boldsymbol{\theta}|\boldsymbol{\xi})}{\partial \xi_i \partial \xi_j}\right\}, \end{aligned} \tag{4}$$

where the last equality is valid given certain regularity conditions (Murray and Rice, 1993).

2.1 COMPUTING THE RIEMANNIAN METRIC TENSOR

When applying natural gradients to approximate inference, the geometry is defined by the approximation $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ and not the full model as usually. If the approximation $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ is chosen such that disjoint groups of variables are independent, that is,

$$q(\boldsymbol{\theta}|\boldsymbol{\xi}) = \prod_i q_i(\boldsymbol{\theta}_i|\boldsymbol{\xi}_i), \tag{5}$$

the computation of the natural gradient is simplified as the Fisher information matrix becomes block-diagonal. The required matrix inversion can be performed very efficiently because

$$\mathrm{diag}(A_1, \ldots, A_n)^{-1} = \mathrm{diag}(A_1^{-1}, \ldots, A_n^{-1}). \tag{6}$$

The dimensionality of the problem space is often so high that inverting the full matrix would not be feasible.

2.2 NORMAL DISTRIBUTION

For the univariate Gaussian distribution parameterized by mean and variance $N(x;\ \mu, v)$, we have

$$\ln q(x|\mu, v) = -\frac{1}{2v}(x-\mu)^2 - \frac{1}{2}\ln(v) - \frac{1}{2}\ln(2\pi). \tag{7}$$

Further,

$$E\left\{-\frac{\partial^2 \ln q(x|\mu, v)}{\partial\mu\partial\mu}\right\} = \frac{1}{v}, \tag{8}$$

$$E\left\{-\frac{\partial^2 \ln q(x|\mu, v)}{\partial v\partial\mu}\right\} = 0, \text{ and} \tag{9}$$

$$E\left\{-\frac{\partial^2 \ln q(x|\mu, v)}{\partial v\partial v}\right\} = \frac{1}{2v^2}. \tag{10}$$

The resulting Fisher information matrix is diagonal and its inverse is given simply by

$$\mathbf{G}^{-1} = \begin{pmatrix} v & 0 \\ 0 & 2v^2 \end{pmatrix}. \tag{11}$$

In the case of univariate Gaussian distribution, natural gradient has a rather straightforward intuitive interpretation as seen in Figure 1. Compared to conventional gradient, natural gradient compensates for the fact that changing the parameters of a Gaussian with small variance has much more pronounced effects than when the variance is large. The differences between the gradient and the natural gradient are illustrated in Figure 2 with a simple example.

For the multivariate Gaussian distribution parameterized by mean and precision $N(\mathbf{x};\ \boldsymbol{\mu}, \boldsymbol{\Lambda})$, we have

$$\begin{aligned} \ln q(\mathbf{x}|\boldsymbol{\mu}, \boldsymbol{\Lambda}) = &-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Lambda}(\mathbf{x}-\boldsymbol{\mu}) \\ &+\frac{1}{2}\log|\det\boldsymbol{\Lambda}| - \frac{d}{2}\ln(2\pi), \end{aligned} \tag{12}$$

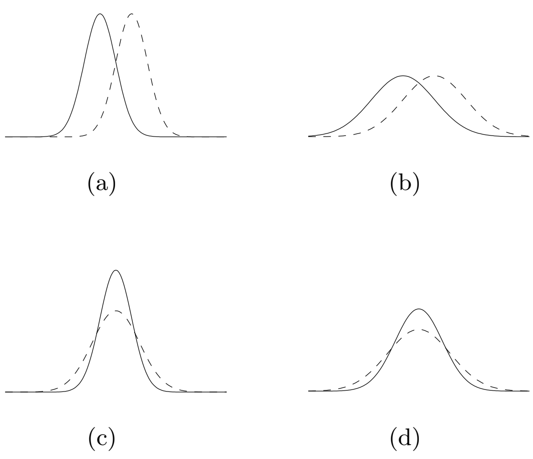

Figure 1: The absolute change in the mean of the Gaussian in figures (a) and (b) and the absolute change in the variance of the Gaussian in figures (c) and (d) is the same. However, the relative effect is much larger when the variance is small as in figures (a) and (c) compared to the case when the variance is high as in figures (b) and (d) (Valpola, 2000).

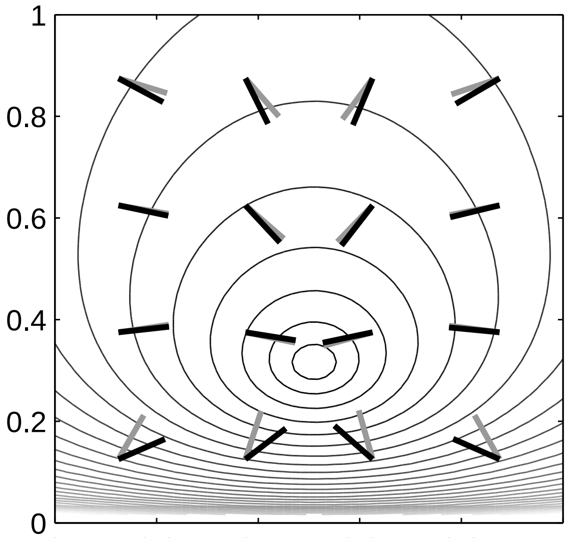

Figure 2: The contours show an objective function of the mean (horizontal axis) and the variance (vertical axis) of a Gaussian model. Gradient (gray line) and natural gradient (black line) are plotted at 16 different points.

where d is the dimension of $\mathbf{x}$. Rather straightforward differentiation yields

$$E\left\{-\frac{\partial^2 \ln q(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Lambda})}{\partial\boldsymbol{\mu}\partial\boldsymbol{\mu}^T}\right\} = \boldsymbol{\Lambda}, \tag{13}$$

$$E\left\{-\frac{\partial^2 \ln q(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Lambda})}{\partial\boldsymbol{\mu}\partial\boldsymbol{\Lambda}}\right\} = 0, \text{ and} \tag{14}$$

$$E\left\{-\frac{\partial^2 \ln q(\mathbf{x}|\boldsymbol{\mu},\boldsymbol{\Lambda})}{\partial\boldsymbol{\Lambda}\partial\boldsymbol{\Lambda}}\right\} = \frac{1}{2}\boldsymbol{\Lambda}^{-1}\otimes\boldsymbol{\Lambda}^{-1}, \tag{15}$$

where $\otimes$ is the direct product, also known as the Kronecker product. Because the cross term is zero, the resulting full Fisher information matrix is block diagonal and can be inverted simply by

$$\mathbf{G}^{-1} = \operatorname{diag}\left(\boldsymbol{\Lambda}^{-1}, 2\boldsymbol{\Lambda}\otimes\boldsymbol{\Lambda}\right). \tag{16}$$

This result for the precision may not be very useful in practice, as the approximations used in most applications have a more restricted form such as a Gaussian with a factor analysis covariance $\boldsymbol{\Sigma} = \mathbf{D} + \sum_{i=1}^{k}\mathbf{v}\mathbf{v}^T$, where $\mathbf{D}$ is a diagonal matrix, or a Gaussian Markov random field.

3 OPTIMIZATION ALGORITHMS ON RIEMANNIAN MANIFOLDS

Many of the traditional optimization algorithms have their direct counterparts in Riemannian space. This paper concentrates on gradient based algorithms, in particular the generalized versions of gradient ascent and conjugate gradient method.

3.1 NATURAL GRADIENT ASCENT

The natural gradient learning algorithm is analogous to conventional gradient ascent algorithm and is given by the iteration

$$\boldsymbol{\xi}_n = \boldsymbol{\xi}_{n-1} + \gamma\tilde{\nabla}\mathcal{F}(\boldsymbol{\xi}_{n-1}), \tag{17}$$

where the step size γ can either be adjusted adaptively during learning or computed for each iteration using e.g. line search (Amari, 1998). This line search should be performed or any longer step taken along a suitable geodesic, which is a length minimizing curve and therefore the Riemannian counterpart of a straight line. In practice, geodesics are often approximated with straight lines (Amari, 1998), as natural gradient ascent is typically applied to problems with complex geometries, and the geodesics on such manifolds can be hard to derive and compute.

In general, the performance of natural gradient learning is superior to conventional gradient learning when the problem space is Riemannian. For instance, natural gradient learning can often avoid the plateaus present in conventional gradient learning (Amari, 1998).

3.2 CONJUGATE GRADIENT METHODS

For better performance it can be useful to combine natural gradient learning with some standard superlinear optimization algorithm. One such algorithm is the nonlinear conjugate gradient (CG) method (Nocedal, 1991). The conjugate gradient method is a standard tool for solving high dimensional nonlinear optimization problems. During each iteration of the conjugate gradient method, a new search direction is generated by conjugation of the residuals from previous iterations. With this choice the search directions form a Krylov subspace and only the previous search direction and the current gradient are required for the conjugation process, making the algorithm efficient in both time and space complexity (Nocedal, 1991).

3.2.1 Riemannian Conjugate Gradient

The extension of the conjugate gradient algorithm to Riemannian manifolds is done by replacing the gradient with the natural gradient. The resulting algorithm is known as the Riemannian conjugate gradient method (Smith, 1993; Edelman et al., 1998). In principle this extension is relatively simple, as it is sufficient that all the vector operations take into account the Riemannian nature of the problem space.

In Riemannian space, the line search should be performed along a geodesic curve, which is the equivalent of Euclidean straight line. Additionally, the old gradient vectors $\tilde{\mathbf{g}}_{k-1}$ defined in a different tangent space should be transformed to the tangent space at the origin of the new gradient by parallel transport along a geodesic (Smith, 1993). The search direction of the Riemannian conjugate gradient algorithm is given by

$$\mathbf{p}_k = \tilde{\mathbf{g}}_k + \beta \tau(\mathbf{p}_{k-1}), \tag{18}$$

where $\tilde{\mathbf{g}}_k = \tilde{\nabla}\mathcal{F}(\boldsymbol{\xi}_k)$ is the natural gradient and $\tau(\mathbf{p}_{k-1})$ is the previous search direction parallelly transported to the current search point. For each iteration, the function is optimized in the search direction using a line search and the iteration is repeated until satisfactory convergence is reached. The multiplier β can be computed in multiple different ways. One popular choice is the Polak-Ribiére formula (Nocedal, 1991; Smith, 1993; Edelman et al., 1998), which in Riemannian space is given by

$$\beta = \frac{(\tilde{\mathbf{g}}_k - \tau(\tilde{\mathbf{g}}_{k-1})) \cdot \tilde{\mathbf{g}}_k}{\tau(\tilde{\mathbf{g}}_{k-1}) \cdot \tilde{\mathbf{g}}_k}, \tag{19}$$

where τ again denotes parallel transport from the previous search point to the current point.

3.2.2 Natural Conjugate Gradient

Like with natural gradient ascent, it is often necessary to make certain simplifying assumptions to keep the iteration simple and efficient. In this paper, geodesics are approximated with (Euclidean) straight lines. This also means that parallel transport cannot be used, and vector operations between vectors from two different tangent spaces are performed in the Euclidean sense, i.e. assuming that the parallel transport between two points close to each other on the manifold can be approximated by the identity mapping. This approximative algorithm is called the natural conjugate gradient (NCG). A similar algorithm was applied to MLP network training by González and Dorronsoro (2006).

For small step sizes and geometries which are locally close to Euclidean these assumptions still retain many of the benefits of original algorithm while greatly simplifying the computations. Edelman et al. (1998) showed that near the solution Riemannian conjugate gradient method differs from the flat space version of conjugate gradient only by third order terms, and therefore both algorithms converge quadratically near the optimum.

The search direction for the natural conjugate gradient method is given by

$$\mathbf{p}_k = \tilde{\mathbf{g}}_k + \beta \mathbf{p}_{k-1}, \tag{20}$$

and the Polak-Ribiére formula is given by

$$\beta = \frac{(\tilde{\mathbf{g}}_k - \tilde{\mathbf{g}}_{k-1}) \cdot \tilde{\mathbf{g}}_k}{\tilde{\mathbf{g}}_{k-1} \cdot \tilde{\mathbf{g}}_k}. \tag{21}$$

4 VARIATIONAL BAYES AND NONLINEAR STATE-SPACE MODELS

Variational Bayesian learning is based on approximating the posterior distribution $p(\boldsymbol{\theta}|\boldsymbol{X})$ with a tractable approximation $q(\boldsymbol{\theta}|\boldsymbol{\xi})$, where $\boldsymbol{X}$ is the data, $\boldsymbol{\theta}$ are the unknown variables (including both the parameters of the model and the latent variables), and $\boldsymbol{\xi}$ are the (variational) parameters of the approximation. The approximation is fitted by maximizing a lower bound on marginal log-likelihood

$$\begin{aligned}\mathcal{B}(q(\boldsymbol{\theta}|\boldsymbol{\xi})) &= \left\langle \log \frac{p(\boldsymbol{X}, \boldsymbol{\theta})}{q(\boldsymbol{\theta}|\boldsymbol{\xi})} \right\rangle \\ &= \log p(\boldsymbol{X}) - D_{\mathrm{KL}}(q(\boldsymbol{\theta}|\boldsymbol{\xi}) \| p(\boldsymbol{\theta}|\boldsymbol{X})),\end{aligned} \tag{22}$$

where $\langle\cdot\rangle$ denotes expectation over q. This is equivalent to minimizing the Kullback–Leibler divergence $D_{\mathrm{KL}}(q\|p)$ between q and p (Ghahramani and Beal, 2001).

4.1 LEARNING ALGORITHMS

Finding the optimal approximation can be seen as an optimization problem, where the lower bound $\mathcal{B}(q(\boldsymbol{\theta}|\boldsymbol{\xi}))$ is maximized with respect to the variational parameters $\boldsymbol{\xi}$. This is often solved using a variational EM algorithm by updating sets of parameters alternatively while keeping the others fixed. Both VE and VM steps can implicitly optimally utilize the Riemannian structure of $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ for conjugate exponential family models (Sato, 2001). Nevertheless, the EM based methods are prone to slow convergence, especially under low noise. A number of methods exist to speed up convergence of EM by more elaborate optimization schemes (McLachlan and Krishnan, 1996; Salakhutdinov et al., 2003) while retaining the alternating structure of E and M steps, but none of these has gained enough popularity to supplant the EM.

The formulation of VB as an optimization problem allows applying generic optimization algorithms to maximize $\mathcal{B}(q(\boldsymbol{\theta}|\boldsymbol{\xi}))$, but this is rarely done in practice because the problems are quite high dimensional. Additionally many of the parameters are in different roles and the lack of this specific knowledge of the geometry of the problem can seriously hinder generic optimization tools.

There exist step lengthening methods that can be used to extend variational EM algorithms such as the pattern search method (Honkela et al., 2003) and adaptive overrelaxation (Salakhutdinov and Roweis, 2003). These methods are easy to implement as they require very little in addition to the EM algorithm. The downside of the methods is the relatively modest speedup, typically only by a small constant factor while the underlying, sometimes linear convergence behavior of variational EM is retained.

4.2 NONLINEAR STATE-SPACE MODEL

As a specific example, let us study the nonlinear state-space model (NSSM) introduced in (Valpola and Karhunen, 2002). The model is specified by the generative model

$$\mathbf{x}(t) = \mathbf{f}(\mathbf{s}(t), \boldsymbol{\theta}_{\mathbf{f}}) + \mathbf{n}(t) \tag{23}$$

$$\mathbf{s}(t) = \mathbf{s}(t-1) + \mathbf{g}(\mathbf{s}(t-1), \boldsymbol{\theta}_{\mathbf{g}}) + \mathbf{m}(t), \tag{24}$$

where t is time, $\mathbf{x}(t)$ are the observations, and $\mathbf{s}(t)$ are the hidden states. The observation mapping $\mathbf{f}$ and the dynamical mapping $\mathbf{g}$ are nonlinear and they are modeled with multilayer perceptron (MLP) networks. Observation noise $\mathbf{n}$ and process noise $\mathbf{m}$ are assumed Gaussian. The latent states $\mathbf{s}(t)$ are commonly denoted by $\boldsymbol{\theta}_{\boldsymbol{S}}$. The model parameters include both the weights of the MLP networks and a number of hyperparameters. The posterior approximation of these parameters is a Gaussian with a diagonal covariance. The posterior approximation of the states $q(\boldsymbol{\theta}_{\boldsymbol{S}}|\boldsymbol{\xi}_{\boldsymbol{S}})$ is also Gaussian, but some dependencies are modeled. The different components of the state vectors are still assumed independent. However, the correlations between the corresponding components of subsequent state vectors $s_j(t)$ and $s_j(t-1)$ are modeled. This is a realistic minimum assumption for modeling the dependence of the state vectors $\mathbf{s}(t)$ and $\mathbf{s}(t-1)$ (Valpola and Karhunen, 2002).

Because of the nonlinearities the model is not in the conjugate exponential family, and the standard VB learning methods are not directly applicable. The bound (22) can nevertheless be evaluated by linearizing the MLP networks $\mathbf{f}$ and $\mathbf{g}$ using the technique of Honkela and Valpola (2005). This allows evaluating the gradient with respect to $\boldsymbol{\xi}_{\boldsymbol{S}}$, $\boldsymbol{\xi}_{\mathbf{f}}$, and $\boldsymbol{\xi}_{\mathbf{g}}$ and using a gradient based optimizer to adapt the parameters. These variables are updated jointly rather than using an EM-like split because the same heavy gradient computations are needed for them all.

The natural gradient with respect to the parameters of $q(\boldsymbol{\theta}_{\boldsymbol{S}}|\boldsymbol{\xi}_{\boldsymbol{S}})$, $q(\boldsymbol{\theta}_{\mathbf{f}}|\boldsymbol{\xi}_{\mathbf{f}})$, and $q(\boldsymbol{\theta}_{\mathbf{g}}|\boldsymbol{\xi}_{\mathbf{g}})$ was simplified by only using the gradient-based updates for the mean elements. For the parameters $q(\boldsymbol{\theta}_{\boldsymbol{S}}|\boldsymbol{\xi}_{\boldsymbol{S}})$ the fully diagonal approximation for the inverse of the metric tensor given by Eqs. (6) and (11) was used. Since the parameters $q(\boldsymbol{\theta}_{\mathbf{f}}|\boldsymbol{\xi}_{\mathbf{f}})$ and $q(\boldsymbol{\theta}_{\mathbf{g}}|\boldsymbol{\xi}_{\mathbf{g}})$ had a diagonal covariance, no further approximations were necessary. Under these assumptions the natural gradient for the mean elements is given by

$$\tilde{\nabla}_{\boldsymbol{\mu}_q} \mathcal{F}(\boldsymbol{\xi}) = \operatorname{diag}(\boldsymbol{\Sigma}_q) \nabla_{\boldsymbol{\mu}_q} \mathcal{F}(\boldsymbol{\xi}), \tag{25}$$

where $\boldsymbol{\mu}_q$ is the mean of the variational approximation $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ and $\operatorname{diag}(\boldsymbol{\Sigma}_q)$ is the diagonal of the corresponding covariance.

Variances were updated separately using a fixed-point update rule as described in (Valpola and Karhunen, 2002). The correlation parameters of $q(\boldsymbol{\theta}_{\boldsymbol{S}}|\boldsymbol{\xi}_{\boldsymbol{S}})$ were updated in EM style by assuming all other parameters fixed. The remaining hyperparameters were updated by VBEM.

5 EXPERIMENTS

As an example, the method for learning nonlinear state-space models presented in Sec. 4.2 was applied to real world speech data. Experiments were made with different data sizes to study the performance differences between the algorithms.

The data set in this experiment was a 21 dimensional real world speech data set. The full data set consisted of 2000 samples of mel frequency log power

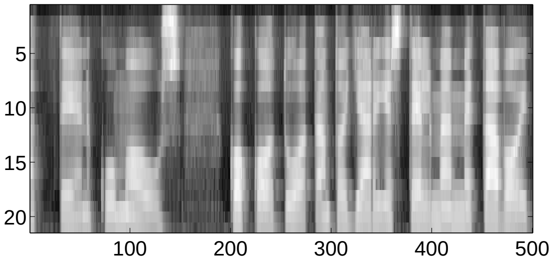

Figure 3: Part of the speech spectrum data used in the experiments.

speech spectra of continuous human speech, which corresponds to roughly 15 seconds of real time. Part of the data set can be seen in Figure 3.

To study the performance differences between the natural conjugate gradient (NCG) method, the conjugate gradient (CG) method and the heuristic algorithm from (Valpola and Karhunen, 2002), the algorithms were applied to different sized parts of the speech data set. Unfortunately a reasonable comparison with a variational EM algorithm was impossible because the extended Kalman smoother (Anderson and Moore, 1979) was unstable and thus the E-step failed.

The size of the data subsets varied between 100 and 500 samples. A five dimensional state-space was used and the MLP networks for the observation and dynamical mapping had 20 hidden nodes. Five different initializations were used to avoid problems with local minima and the results were averaged over the different iterations. An iteration was assumed to have converged when $|\mathcal{B}^t - \mathcal{B}^{t-1}| < (5 \cdot 10^{-3}/N)$ for 200 consecutive iterations, where $\mathcal{B}^t$ is the bound on marginal log-likelihood at iteration t and N is the size of the data set.

The results can be seen in Figure 4. In particular, as the data size increases, natural conjugate gradient tends to perform much better than the competing algorithms. The slightly anomalous behavior at the data size of 200 can be explained by a silent period in the speech data set between samples 150 and 200.

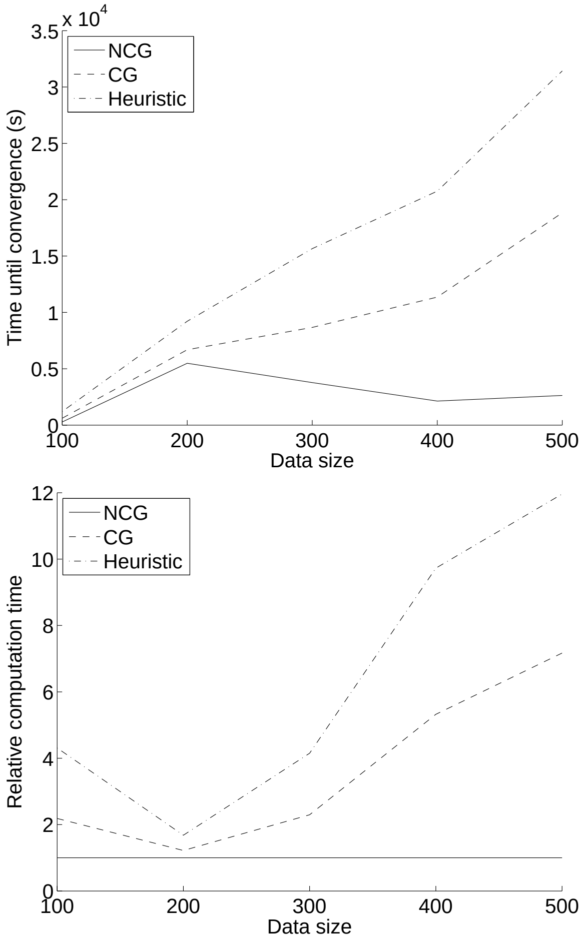

Figure 4: Convergence speed of the natural conjugate gradient (NCG) method, the conjugate gradient (CG) method and the heuristic algorithm with different data sizes. Top: Absolute computation times. Bottom: Relative computation times with the computation time of NCG method normalized to 1.

The difference in the performance of the algorithms can be at least partially explained by the fact that the ratios of the variances of the different parameters change as the data size increases. The variance of the dynamical and observation mapping weights will tend to get smaller as the data size increases, but there will always be uncertainty left in the states. The variances of Gaussian distributions scale the natural gradient as seen in Eq. (25). Therefore large relative difference in variances can help to explain the poor performance of methods based on flat geometry with larger data sets, as the corrections imposed by the Riemannian geometry become more significant. The effect of data size on the variances is illustrated in Figure 5, where the ratio of the minimum of the normalized variances of the states and observation network output weights is plotted against data size.

As a slightly more realistic example, the full data set of 2000 samples was used to train a seven dimensional state-space model. In this experiment both MLP networks of the NSSM had 30 hidden nodes.

The performance of the NCG method, CG method and the heuristic algorithm was compared. The results can be seen in Figure 6. Five different initializations were used to avoid problems with poor local optima. The

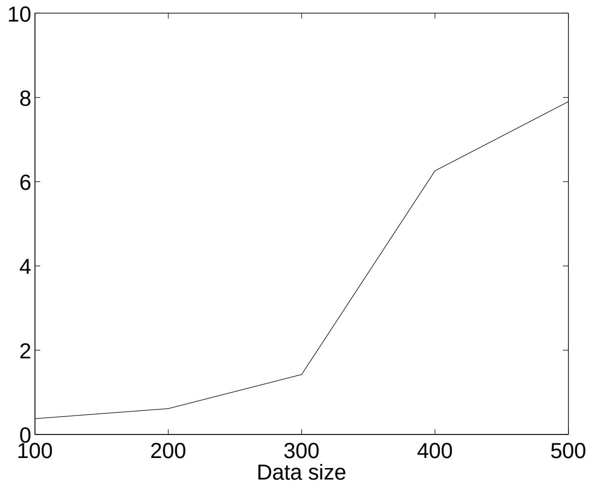

Figure 5: Ratio of the normalized posterior variance of the states and the observation network output layer weights after the iteration has converged. The results are averaged over the different methods, as they all produced similar results.

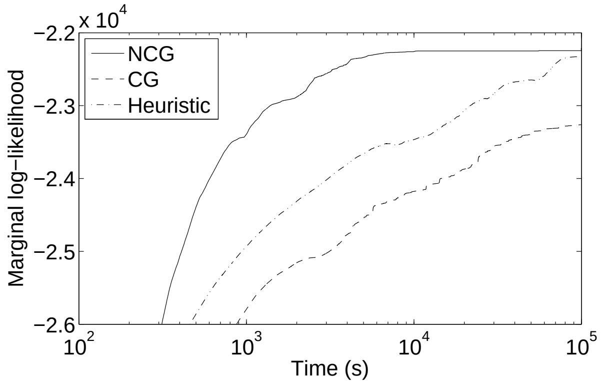

Figure 6: Comparison of the performance of the natural conjugate gradient (NCG) method, the conjugate gradient (CG) method and the heuristic algorithm with the full data set. Lower bound on marginal log-likelihood $\mathcal{B}$ is plotted against computation time.

results presented in Figure 6 are from the iterations that converged to the best local optimum.

Natural conjugate gradient clearly outperformed the other algorithms in this experiment. In particular, conventional conjugate gradient learning converged very slowly with this larger data set and regardless of initialization failed to reach a local optimum within reasonable time. Natural conjugate gradient also outperformed the heuristic algorithm (Valpola and Karhunen, 2002) by a factor of more than 10.

6 DISCUSSION

In previous machine learning algorithms based on natural gradients (Amari, 1998), the aim has been to use maximum likelihood to directly update the model parameters $\boldsymbol{\theta}$ taking into account the geometry imposed by the predictive distribution for data $p(\boldsymbol{X}|\boldsymbol{\theta})$. The resulting geometry is often much more complicated as the effects of different parameters cannot be separated and the Fisher information matrix is relatively dense.

In this paper, only the simpler geometry of the approximating distributions $q(\boldsymbol{\theta}|\boldsymbol{\xi})$ is used. Because the approximations are often chosen to minimize dependencies between different parameters $\boldsymbol{\theta}$, the resulting Fisher information matrix with respect to the variational parameters $\boldsymbol{\xi}$ will be mostly diagonal and hence easy to invert.

While taking into account the structure of the approximation, plain natural gradient in this case ignores the structure of the model and the global geometry of the parameters $\boldsymbol{\theta}$. This is to some extent addressed by using conjugate gradients, and even more sophisticated optimization methods such as quasi-Newton or even Gauss–Newton methods can be used if the size of the problem permits it.

While the natural conjugate gradient method has been formulated mainly for models outside the conjugate-exponential family, it can also be applied to conjugate-exponential models instead of the more common variational EM algorithms. In practice, simpler and more straightforward EM acceleration methods may still provide comparable results with less human effort.

The experiments in this paper show that even a diagonal approximation of the Riemannian metric tensor is enough to acquire a large speedup. Considering univariate Gaussian distributions, the regular gradient is too strong for model variables with small posterior variance and too weak for variables with large posterior variance, as seen from Equations (8)–(10). The posterior variance of latent variables is often much larger than the posterior variance of model parameters, which means that maximal benefit from the natural gradient can be attained by combining at least parts of E and M steps of the variational EM.

When the data set is small, regular conjugate gradient method works reasonably well. However, for larger data sets natural conjugate gradient shows far superior performance.

Initial experiments with natural gradient ascent (without conjugacy) indicated that its performance is significantly worse than the other compared algorithms. However, it is possible that natural gradient ascent suffers more than natural conjugate gradient method from the approximations made in the computation of the Riemannian metric tensor.

7 CONCLUSION

We have presented a novel method to speed up learning methods based on optimizing an objective function depending on a probability distribution, such as variational Bayesian learning. Taking into the account the Riemannian structure among the variational parameters, the natural conjugate gradient algorithm is efficiently used to update both latent variables and model parameters at the same time. A simple form of the approximate distribution translates into a simple metric and a low computational overhead, but even for more complicated approximating distributions a simple approximation of the metric can provide significant speedups at very low extra computational cost.

Acknowledgments

This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

References

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

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

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

Anderson, B. and Moore, J. (1979). *Optimal Filtering*. Prentice-Hall, Englewood Cliffs, NJ.

Barber, D. and Bishop, C. (1998). Ensemble learning for multi-layer networks. In Jordan, M., Kearns, M., and Solla, S., editors, *Advances in Neural Information Processing Systems 10*, pages 395–401. The MIT Press, Cambridge, MA, USA.

Edelman, A., Arias, T. A., and Smith, S. T. (1998). The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353.

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

González, A. and Dorronsoro, J. R. (2006). A note on conjugate natural gradient training of multilayer perceptrons. In *Proc. Int. Joint Conf. on Neural Networks (IJCNN'06)*, pages 887–891, Vancouver, BC, Canada.

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

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

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

McLachlan, G. J. and Krishnan, T. (1996). *The EM Algorithm and Extensions*. Wiley.

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

Nocedal, J. (1991). Theory of algorithms for unconstrained optimization. *Acta Numerica*, 1:199–242.

Salakhutdinov, R. and Roweis, S. T. (2003). Adaptive overrelaxed bound optimization methods. In *Proc. 20th International Conference on Machine Learning (ICML 2003)*, pages 664–671.

Salakhutdinov, R., Roweis, S. T., and Ghahramani, Z. (2003). Optimization with EM and expectation-conjugate-gradient. In *Proc. 20th International Conference on Machine Learning (ICML 2003)*, pages 672–679.

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

Seeger, M. (2000). Bayesian model selection for support vector machines, Gaussian processes and other kernel classifiers. In Solla, S., Leen, T., and Müller, K.-R., editors, *Advances in Neural Information Processing Systems 12*, pages 603–609. MIT Press, Cambridge, MA, USA.

Smith, S. T. (1993). *Geometric Optimization Methods for Adaptive Filtering*. PhD thesis, Harvard University, Cambridge, Massachusetts.

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

Valpola, H. (2000). *Bayesian Ensemble Learning for Nonlinear Factor Analysis*. PhD thesis, Helsinki University of Technology, Espoo, Finland. Published in Acta Polytechnica Scandinavica, Mathematics and Computing Series No. 108.

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

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

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

Yang, H. H. and Amari, S. (1997). Adaptive online learning algorithms for blind separation: Maximum entropy and minimum mutual information. *Neural Computation*, 9(7):1457–1482.