NATURAL GRADIENT MATRIX MOMENTUM

Silvia Scarpetta[+,*], Magnus Rattray[†] and David Saad[*]

[+]Physics Department "E. R. Caianiello", Salerno University, Baronissi, (SA), IT and INFM, Sezione di Salerno, (SA), IT
[†]Computer Science Department, University of Manchester, M13 9PL, UK
[*]Neural Computing Research Group, Aston University, Birmingham B4 7ET, UK

ABSTRACT

Natural gradient learning is an efficient and principled method for improving on-line learning. In practical applications there will be an increased cost required in estimating and inverting the Fisher information matrix. We propose to use the matrix momentum algorithm in order to carry out efficient inversion and study the efficacy of a single step estimation of the Fisher information matrix. We analyse the proposed algorithms in a two-layer neural network, using a statistical mechanics framework which allows us to describe analytically the learning dynamics, and compare performance with true natural gradient learning and standard gradient descent.

1 INTRODUCTION

On-line learning is a popular method for training multi-layer feed-forward neural networks, where network parameters are updated according to only the latest in a sequence of training examples. On-line methods can be beneficial in terms of both storage and computation time, and also allow for temporal changes in the task being learned. An overview of on line learning methods in neural networks can be found in [1].

Natural gradient learning [2] was recently proposed by Amari as a more principled alternative to standard on-line gradient descent. The natural gradient method makes use of the Riemannian metric given by the Fisher information matrix to optimise the learning dynamics. The idea is to convert the covariant gradient into the contravariant form, obtained by pre-multiplying the standard gradient with the inverse of the Fisher information matrix. Natural gradient learning is proved to be Fisher efficient [2], implying that it has asymptotically the same performance as the optimal batch estimation of parameters; moreover, this learning algorithm can also provide improved performance over standard gradient descent during the transient stages of learning, with improved scaling of learning time against task complexity [3]. In practical applications there will be an increased cost required in estimating and inverting the Fisher information matrix. Determining this matrix on-line is difficult as we require an average over the distribution of inputs in order to calculate it. Even if the Fisher information matrix can be computed on-line, inverting it will be computationally costly when our network is large. This is particularly undesirable when we consider that computational efficiency is one of the principal reasons for using on-line methods. An on-line matrix momentum (MM) algorithm [4, 5] was recently introduced in order to invert an estimate of the Hessian efficiently on-line. We propose to use this method to compute the inverse of the Fisher information matrix as required for natural gradient learning. This method is particularly efficient since the inversion is replaced by a matrix-vector multiplication which can be carried out by a back-propagation step [4].

The aim of this paper is twofold. First, we employ a theoretical framework, recently developed [6] for studying the learning dynamics of on-line learning in order to study the performance of MM using the averaged Fisher information matrix. Second, we use the same theoretical framework to examine performance using a single pattern estimation of the Fisher information matrix.

2 NATURAL GRADIENT

Consider a mapping from an input space $\boldsymbol{\xi} \in \mathbb{R}^N$ onto a scalar $\phi_{\mathbf{J}}(\boldsymbol{\xi}) = \sum_{i=1}^{K} g(\mathbf{J}_i^{\mathrm{T}}\boldsymbol{\xi})$, which defines a soft committee machine (we call this the 'student' network), where $\mathbf{J} \equiv$

scarpetta@na.infn.it, magnus@cs.man.ac.uk, saadd@aston.ac.uk

$\{\mathbf{J}_i\}_{1\leq i\leq K}$ is the set of input to hidden weights and the hidden to output weights are set to one. We choose $g(x) \equiv \mathrm{erf}(x/\sqrt{2})$ to be the activation function of the hidden units. We can then define a Gaussian noise model for output ζ_m given input $\boldsymbol{\xi}$ which is parameterised by $\mathbf{J}$,

$$p_\mathbf{J}(\zeta_\mathrm{m}|\boldsymbol{\xi}) = \frac{1}{\sqrt{2\pi\sigma_\mathrm{m}^2}} \exp\left(\frac{-(\zeta_\mathrm{m} - \phi_\mathbf{J}(\boldsymbol{\xi}))^2}{2\sigma_\mathrm{m}^2}\right) . \tag{1}$$

Let $(\boldsymbol{\xi}^\mu, \zeta^\mu)$ be the μ-th input-output pair in a sequence of training examples. The activation of the student hidden node i under presentation of the input pattern $\boldsymbol{\xi}^\mu$ is denoted $x_i^\mu = \mathbf{J}_i^\mathrm{T}\boldsymbol{\xi}^\mu$. The training error at each learning iteration is taken to be proportional to the log-likelihood of the current example under our noise model, $\epsilon_\mathbf{J}(\zeta^\mu, \boldsymbol{\xi}^\mu) \equiv \frac{1}{2}(\zeta^\mu - \sum_{i=1}^K g(x_i^\mu))^2$ and the most basic learning algorithm is to adapt the student weights in the negative gradient direction of this error at each iteration. A more principled learning algorithm can be defined by viewing the manifold of models as a Reimannian space in which local distance is defined by the KL-divergence. The Fisher information matrix $\mathbf{G}$ defines the appropriate metric in this space [7] ,

$$\mathbf{G} = \left\langle (\nabla_\mathbf{J} \log p_\mathbf{J}(\zeta_\mathrm{m}|\boldsymbol{\xi}))(\nabla_\mathbf{J} \log p_\mathbf{J}(\zeta_\mathrm{m}|\boldsymbol{\xi}))^\mathrm{T} \right\rangle , \tag{2}$$

where the brackets denote an average over ζ_m, according to equation (1), followed by an average over the input distribution. Let $\mathbf{G}_{ik}$ be block (i,k) of the Fisher information, which for our network is:

$$\mathbf{G}_{ik} = \frac{1}{\sigma_\mathrm{m}^2} \langle \mathbf{A}_{ik}(\boldsymbol{\xi}) \rangle_{\{\xi\}} \tag{3}$$

where

$$\mathbf{A}_{ik}(\boldsymbol{\xi}) = g'(x_i^\mu) g'(x_k^\mu) \boldsymbol{\xi}\boldsymbol{\xi}^\mathrm{T} \tag{4}$$

The natural gradient direction is found by pre-multiplying the training error gradient by the inverse of this matrix. When components of $\boldsymbol{\xi}$ are selected independently at each iteration from a zero-mean Gaussian distribution with unit variance, the matrix $\mathbf{G}$ can be computed analytically, and, for our particular choice of activation function, we find, [3]

$$\langle \mathbf{A}_{ik}(\boldsymbol{\xi}) \rangle_{\{\xi\}} = \frac{2}{\pi\sqrt{\Delta_{ik}}} \left\{ \mathbf{I} - \frac{1}{\Delta_{ik}} \left[(1+Q_{kk})\mathbf{J}_i\mathbf{J}_i^\mathrm{T} + (1+Q_{ii})\mathbf{J}_k\mathbf{J}_k^\mathrm{T} - Q_{ik}(\mathbf{J}_i\mathbf{J}_k^\mathrm{T} + \mathbf{J}_k\mathbf{J}_i^\mathrm{T}) \right] \right\} \tag{5}$$

with $Q_{ik} \equiv \mathbf{J}_i^\mathrm{T}\mathbf{J}_k$ and $\Delta_{ik} = (1+Q_{ii})(1+Q_{kk}) - Q_{ik}^2$. When the pdf of the input is unknown, we should estimate the average $\langle \mathbf{A}_{ik}(\boldsymbol{\xi}) \rangle_{\{\xi\}}$ on the basis of an empirically estimated input distribution. However, it is difficult to implement the natural gradient descent method as an on-line algorithm in this way. In some cases it will be possible to estimate the input pdf on-line, and Yang & Amari [7] discuss methods of preprocessing training examples to obtain a whitened Gaussian process for the inputs in this case. For $N \gg K$ this gives efficient inversion, but this will not be possible in general.

3 GENERAL FRAMEWORK

We use a statistical mechanics description of the learning process [6] which is exact in the limit of large input dimension N and provides an accurate model of mean behaviour for realistic values of N. Training examples are of the form $(\boldsymbol{\xi}^\mu, \zeta^\mu)$, as introduced in the previous section, where $\mu = 1, 2, \ldots$ labels each independently drawn example in a sequence. The output ζ^μ is given by a teacher which may be corrupted by output noise and is of a similar configuration to the student except for a possible difference in the number M of hidden units: $\zeta^\mu = \sum_{n=1}^M g(\mathbf{B}_n^\mathrm{T}\boldsymbol{\xi}^\mu) + \rho^\mu$, where $\mathbf{B} \equiv \{\mathbf{B}_n\}_{1\leq n\leq M}$ is the set of input to hidden adaptive weights for teacher hidden nodes and ρ^μ is zero-mean Gaussian noise of variance σ^2. Due to the flexibility of this teacher mapping we can represent a variety of learning scenarios within this theoretical framework [8]. The activation of hidden node n in the teacher under presentation of the input pattern $\boldsymbol{\xi}^\mu$ is denoted $y_n^\mu = \mathbf{B}_n^\mathrm{T}\boldsymbol{\xi}^\mu$.

In the natural gradient algorithm the weight update at each iteration is given by:

$$\mathbf{J}_i^{\mu+1} = \mathbf{J}_i^\mu + \frac{\eta}{N} \sum_{j=1}^K \delta_j^\mu [\langle \mathbf{A}(\boldsymbol{\xi}) \rangle_{\{\xi\}}^{-1}]_{ij} \boldsymbol{\xi}^\mu , \tag{6}$$

where

$$\delta_i^\mu \equiv g'(x_i^\mu) \Big[\sum_{n=1}^M g(y_n^\mu) - \sum_{j=1}^K g(x_j^\mu) + \rho^\mu \Big]$$

and the learning rate η has been scaled with the input size N. Notice that knowl-

edge of the teacher noise variance is not required to execute this algorithm. Assuming a Gaussian input distribution it is then possible to derive equations of motion, for both gradient descent [6] and natural gradient learning [3], for a set of order parameters $\langle x_i x_j \rangle = \mathbf{J}_i^{\mathrm{T}} \mathbf{J}_k \equiv Q_{ik}$, $\langle x_i y_n \rangle = \mathbf{J}_i^{\mathrm{T}} \mathbf{B}_n \equiv R_{in}$, and $\langle y_n y_m \rangle = \mathbf{B}_n^{\mathrm{T}} \mathbf{B}_m \equiv T_{nm}$, measuring overlaps between student and teacher vectors. These order parameters are necessary and sufficient to determine the generalisation error $\epsilon_g = \langle \epsilon_{\mathbf{J}}(\zeta^\mu, \boldsymbol{\xi}^\mu) \rangle_{\{\xi\}}$. The equations of motion are in the form of coupled first order differential equations for the order parameters with respect to the normalised number of examples $\alpha = \mu/N$ and we can integrate them numerically in order to determine the evolution of the generalisation error. In the following sections we will show how the inclusion of an extra set of macroscopics allows equations to also be determined for natural gradient MM.

4 NATURAL GRADIENT MATRIX MOMENTUM

A heuristic which is sometimes useful in batch learning is to include a momentum term in the basic gradient descent algorithm. For on-line gradient descent learning with momentum we have,

$$\mathbf{J}_i^{\mu+1} = \mathbf{J}_i^{\mu} + \frac{\eta}{N} \delta_i^\mu \boldsymbol{\xi}^\mu + \beta(\mathbf{J}_i^{\mu} - \mathbf{J}_i^{\mu-1}) \ . \quad (7)$$

An interesting behaviour is observed if we choose $\eta \sim O(1/N)$ and $(1-\beta) \sim O(1/N)$ [9, 4]. If we define $\eta = k/N$ and $\beta = 1 - \gamma/N$, then taking $\gamma \to \infty$ and $k \to \infty$ simultaneously while keeping their ratio finite results in dynamics equivalent to gradient descent with an effective learning rate of $\eta_{\mathrm{eff}} = k/\gamma$. If we choose a matrix momentum parameter

$$\boldsymbol{\beta} = \mathbf{I} - \frac{k\mathbf{A}}{N} \quad \text{with} \quad \eta = \frac{k\eta_\alpha}{N} \ , \quad (8)$$

one might then expect that the learning rate rescaling described above results in an effective matrix learning rate $\eta_{\mathrm{eff}} = \eta_\alpha \mathbf{A}^{-1}$ (see [5] for an analysis of the dynamics in the case where $\mathbf{A}$ is the Hessian matrix). We will choose $\mathbf{A}$ proportional to the Fisher information matrix, so that by making k large we expect to retrieve natural gradient learning. In order to solve the dynamics we define a new set of variables $\boldsymbol{\Omega}_i^\mu = N(\mathbf{J}_i^\mu - \mathbf{J}_i^{\mu-1})$, so that the learning step can be described as a first order process:

$$\mathbf{J}_i^{\mu+1} = \mathbf{J}_i^{\mu} + \frac{\eta}{N} \delta_i^\mu \boldsymbol{\xi}^\mu + \frac{1}{N} [\boldsymbol{\beta}\boldsymbol{\Omega}]_i^\mu \quad (9)$$

$$\boldsymbol{\Omega}_i^{\mu+1} = [\boldsymbol{\beta}\boldsymbol{\Omega}]_i^\mu + \eta \, \delta_i^\mu \boldsymbol{\xi}^\mu \ . \quad (10)$$

This gives rise to a new set of Gaussian fields related to the new variable: $z_i^\mu = \boldsymbol{\Omega}_i^{\mathrm{T}} \boldsymbol{\xi}^\mu$. As in [9], we define a new set of order parameters relating the new momentum variables: $\langle z_i z_k \rangle = \boldsymbol{\Omega}_i^{\mathrm{T}} \boldsymbol{\Omega}_k \equiv C_{ik}$, $\langle z_i y_n \rangle = \boldsymbol{\Omega}_i^{\mathrm{T}} \mathbf{B}_n \equiv D_{in}$, and $\langle x_i z_k \rangle = \mathbf{J}_i^{\mathrm{T}} \boldsymbol{\Omega}_k \equiv E_{ik}$. In the following sections we derive and discuss the differential equations describing the evolution of the order parameters, both for the true averaged Fisher information matrix and for a crude on-line estimate.

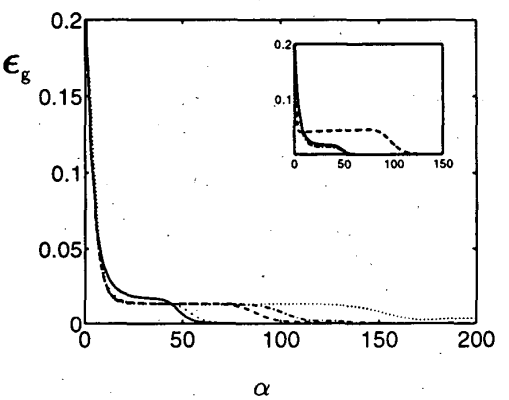

Figure 1: The solid lines show the generalisation error for natural gradient descent ($\eta = 0.15$) for a two hidden node network learning a noiseless task. We compare it with the result for MM using the exact Fisher information matrix, with $\eta_{\mathrm{eff}} = 0.15$ and $k = 0.5$ (dotted), $k = 1.4$ (dot-dash), $k = 2.1$(dashed), $k = 10$ (dots). The inset shows the optimal gradient descent result (dashed), natural gradient descent (solid), and MM (dot-dash) with $k = 20$. Task is isotropic with $T_{mn} = \delta_{mn}$, and initial conditions are $Q_{i \neq k}, R_{in} = U[0, 10^{-3}]$, $Q_{ii} = U[0, 0.5]$, $C_{ik} = D_{in} = E_{ik} = 0$.

4.1 MM with averaged Fisher information matrix

In order to obtain dynamics equivalent to natural gradient descent we choose $\mathbf{A} = \langle \mathbf{A}_{ij}(\boldsymbol{\xi}) \rangle_{\{\xi\}}$, as given in eqn. (5). Along

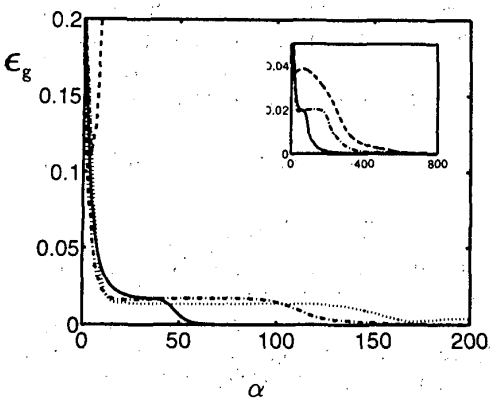

Figure 2: The solid lines show the generalisation error for natural gradient descent ($\eta = 0.15$) for a two hidden node network learning a noiseless task. We show generalisation error of MM using a single pattern estimate of the Fisher information matrix, with $\eta_{eff} = 0.15$, and $k = 0.5$ (dotted), $k = 1.4$ (dot-dash), $k = 2.1$(dashed). The inset shows the optimal gradient descent w.r.t. η (dashed), the natural gradient descent (solid) with $\eta = 0.08$ annealed from the end of the symmetric phase, and MM using the on-line estimated Fisher information (dash-dot) with $k = 1$ and $\eta_{\text{eff}} = 0.08$ annealed from the end of the symmetric phase, for a two node network learning a noisy task ($\sigma^2 = 0.1$). Task is isotropic with $T_{mn} = \delta_{mn}$, and initial conditions are $Q_{i\neq k}, R_{in} = U[0, 10^{-3}]$, $Q_{ii} = U[0, 0.5]$, $C_{ik} = D_{in} = E_{ik} = 0$.

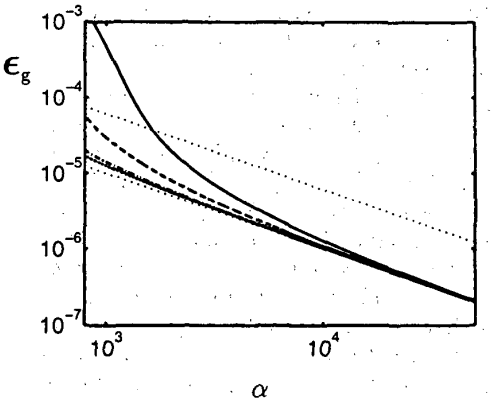

Figure 3: Asymptotic performance of natural gradient MM using the true Fisher information matrix, with $k = 0.3, 0.7, 1.4, 10$ in descending order. The dotted lines show the optimal gradient descent asymptotic decay (upper dotted line), and the universal batch asymptotic bound (lower dotted line), for a two node network learning an isotropic task, with $T_{mn} = \delta_{mn}/2$, in presence of noise ($\sigma^2 = 0.01$).

the lines of [6] we derive a closed set of differential equations describing the evolution of the order parameters,

$$\begin{aligned}
\frac{dQ_{ik}}{d\alpha} &= E_{ik} + E_{ki} \,, \\
\frac{dR_{in}}{d\alpha} &= D_{in} \,, \\
\frac{dC_{ik}}{d\alpha} &= -k\Omega_i^T \textstyle\sum_j A_{kj}\Omega_j - k\sum_j \Omega_j^T A_{ij}\Omega_k \\
&\quad + k\eta_\alpha \langle \delta_k z_i + \delta_i z_k \rangle + k^2\eta_\alpha^2 \langle \delta_i \delta_k \rangle \,, \\
\frac{dD_{in}}{d\alpha} &= -k\textstyle\sum_j \Omega_j^T A_{ij} B_n + k\eta_\alpha \langle \delta_i y_n \rangle \,, \\
\frac{dE_{ik}}{d\alpha} &= C_{ik} - k\textstyle\sum_j \Omega_j^T A_{kj} J_i + k\eta_\alpha \langle \delta_k x_i \rangle \,,
\end{aligned}$$

where the angled brackets denote averages over inputs, or equivalently averages over the field variables $\{x_i\}$, $\{y_n\}$ and $\{z_i\}$, which can be carried out analytically, and

$$\textstyle\sum_j A_{ij}\Omega_j = \frac{2}{\pi}\sum_j \frac{1}{\sqrt{\Delta_{ij}}}\left[\Omega_j - \frac{(1+Q_{jj})E_{ij}J_i}{\Delta_{ij}} + \frac{(1+Q_{ii})E_{jj}J_j - Q_{ij}(E_{jj}J_i + E_{ij}J_j)}{\Delta_{ij}}\right] .$$

In Fig. 1 we compare the performance of the MM method to the natural gradient learning in which the Fisher matrix has been inverted explicitly [3], for a two-node network learning a noiseless isotropic task. The dotted and/or dashed lines show results for $k = 0.5$, $k = 1.4$, $k = 2.1$ and $k = 10$, from right to left. As k increases, the trajectory converges close to the natural gradient learning result (solid line). For comparison we show in the insert the optimal gradient descent result, where the optimal time-dependent learning rate is determined by maximising the total change in generalisation error by a variational approach , as described in [10]. It is well known that natural gradient learning is asymptotically optimal in presence of output noise with annealed learning rate $\eta = 1/\alpha$, equalling in performance the best possible batch algorithm. In Fig. 3 we show the asymptotic performance of natural gradient MM for a two node network learning an isotropic task in presence of noise ($\sigma^2 = 0.01$). The asymptotic result for natural gradient learning takes a very simple form: $\epsilon_g \sim K\sigma^2/2\alpha$ [3] (lower dotted line), equalling the universal batch asymptotics for optimal Bayes and maximum like-

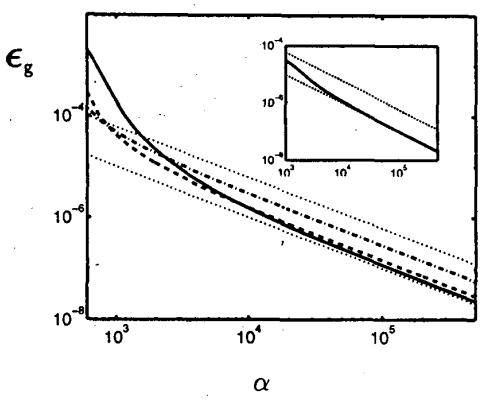

Figure 4: Asymptotic performance of natural gradient MM using an on-line estimate of the Fisher information matrix, with $k = 0.3$ (solid), $k = 0.7$ (dashed) and $k = 1.4$ (dot-dash). The dotted lines show the optimal gradient descent asymptotic decay (upper dotted line), and the universal batch asymptotic bound (lower dotted line), for a two node network learning an isotropic task, with $T_{mn} = \delta_{mn}/2$, in presence of noise ($\sigma^2 = 0.01$). In the inset we show the asymptotic decay when k is annealed from $\alpha > \alpha_o \equiv 800$ as $k = k_o/(1+k_o \log(\alpha - \alpha_o + 1))$.

lihood predictors [11]. The best asymptotic performance that is possible to obtain with gradient descent, by an appropriate annealing of the learning rate, was determined in [12] (upper dotted line). We see that natural gradient matrix momentum also saturates the universal bounds on asymptotic performance, equalling the natural gradient asymptotic decay independent from the choice of k, providing a significant improvement even over optimised gradient descent. Matrix momentum therefore provides an efficient approximation to natural gradient method when the Fisher information matrix is known. In the next section we consider MM algorithm with an on-line approximation to the Fisher information.

4.2 MM with single pattern Fisher information

In order to define a practical on-line algorithm in the case when the input distribution is unknown and far from being Gaussian, we need some approximation to the Fisher information which can be determined on-line. The simplest such approximation is to use a single training example estimation; that is we no longer average the expressions in eqs. (2) and (3). Using $\mathbf{A} = [\mathbf{A}_{ij}(\boldsymbol{\xi})]$ we find the same equations for $\frac{dQ_{ik}}{d\alpha}$ and $\frac{dR_{in}}{d\alpha}$, and for the other order parameters we find:

$$\begin{aligned}
\frac{dC_{ik}}{d\alpha} &= k\langle(\eta_\alpha\delta_i - \phi_i)z_k + (\eta_\alpha\delta_k - \phi_k)z_i\rangle \\
&\quad + k^2\langle(\eta_\alpha\delta_i - \phi_i)(\eta_\alpha\delta_k - \phi_k)\rangle \ , \\
\frac{dD_{in}}{d\alpha} &= k\langle(\eta_\alpha\delta_i - \phi_i)y_n\rangle \ , \\
\frac{dE_{ik}}{d\alpha} &= C_{ik} + k\langle(\eta_\alpha\delta_k - \phi_k)x_i\rangle \ . \qquad (11)
\end{aligned}$$

Here, we have defined $\phi_i = g'(x_i)\sum_j z_j g'(x_j)$. All averages can be carried out explicitly to provide a closed set of equations of motion.

In Fig. 2 we examine the dependence of this single pattern estimation method to the choice of k. We see that choosing k too large ($k = 2.1$, dash line) may lead to potentially uncontrolled behaviour, (due to fluctuations in the single pattern estimate) and k too small ($k = 0.5$, dotted line) may lead to a long transient time. As we approach intermediate values of k we obtain good performance ($k = 1.4$, dash-dot line), that, especially in noisy and overealizable tasks, provides an improvement over gradient descent. In the inset we compare the single pattern MM natural gradient method ($k = 0.8$, $\eta_{\text{eff}} = 0.1$, dash-dot line) with optimal gradient descent (dotted line), and natural gradient learning ($\eta = 0.1$, solid line), for a noisy task, showing a reduction in learning time over gradient descent but not equalling the performance of natural gradient learning. In Fig. 4 we compare the generalisation error asymptotic decay for natural gradient descent (lower bound), optimal gradient descent (upper dotted line) and the single pattern MM for various values of k. We see that the prefactor of the asymptotic decay for single pattern MM increases when k increases. We suspect that this is due to the strong fluctuations in the single pattern estimate, enhanced by large values of k. As k decreases, the asymptotic decay converges close to the optimal bound, but it takes longer to reach this asymptotic regime. In the inset we show that annealing k results in a trajectory which converges rapidly to the optimal bound. Fur-

ther work is required to determine the optimal and maximal values of k and η_α analytically, using methods from [12] , since we have shown here that performance is strongly dependent on the parameter choice.

5 CONCLUSIONS

The natural gradient learning algorithm is efficient, and provides a significant improvement over conventional on-line training methods; however, its complexity is generally high due to the computation required for inverting the Fisher information matrix. Here we have shown that an efficient inversion may be achieved using the matrix momentum algorithm. We also exploited the theoretical framework to study the efficiency of the single pattern estimation of the Fisher information matrix. It turns out that good performance is still possible but with some sensitivity to parameter choice, due to noise in the Fisher information estimate. It will be essential to consider more sophisticated on-line approximations to the Fisher information, which might provide greater robustness to the choice of parameters. The present formalism provides an ideal theoretical framework in which to consider such adaptations.

Acknowledgements

Support from EPSRC grant GR/L19232 is gratefully acknowledged. We would also like to thank S-J Farmer for contribution to Figures 1 and 3.

6 References

[1] Saad D (ed) 1998 *On-Line Learning in Neural Networks*, Publications of the Newton Institute, Cambridge University Press, Cambridge.

[2] Amari S 1998 'Natural gradient works efficiently in learning' *Neural Comp.* **10** 251.

[3] Rattray M and Saad D 1998 'Transient and Asymptotics od Natural Gradient Learning', *Proc. of the Int. Conf. on Artificial Neural Networks*, ed Niklasson, Bodòden and Ziemke (London, UK: Springer-Verlag) 165; Rattray M Saad D and Amari S 1998, *Phys. Rev. Lett.*, **81** 5461; Rattray M and Saad D 1999, *Phys. Rev.* E, in press.

[4] Orr G B and Leen T K 1997 *Advances in Neural Information Processing Systems*, vol **9**, ed Mozer, Jordan and Petsche (Cambridge, MA: MIT Press) 606.

[5] Rattray M and Saad D 1998 'The Dynamics of Matrix Momentum', *Proc. of the Int. Conf. on Artificial Neural Networks*, ed Niklasson, Bodòden and Ziemke (London, UK: Springer-Verlag) 183.

[6] Saad D and Solla S A 1995 *Phys. Rev. Lett.* **74** 4337, *Phys. Rev.* E **52** 4225.

[7] Yang H Y and Amari S 1997 *Advances in Neural Information Processing Systems*, vol **10**, ed Mozer, Jordan and Petsche (Cambridge, MA: MIT Press) 385.

[8] Cybenko *Math. Control Signals and Systems* **2** 303.

[9] Prügel-Bennett A 1996, unpublished notes.

[10] Saad D and Rattray M 1997 *Phys. Rev. Lett.* **79** 2578; Rattray M and Saad D 1998 *Phys. Rev.* E **58** 6379.

[11] Amari S and Murata N 1993, *Neural Comp.* **5** 140.

[12] Leen T K, Schottky B and Saad D 1998 *Advanced in Neural Information Systems* Vol. **10**, 301, ed. by Jordan, Kearns, Solla (Cambridge, MA: MIT Press); Leen T K, Schottky B and Saad D 1999 *Phys. Rev.* E **59** 985.