Approximated Geodesic Updates with Principal Natural Gradients

Zhirong Yang and Jorma Laaksonen

***Abstract*— We propose a novel optimization algorithm which overcomes two drawbacks of Amari's natural gradient updates for information geometry. First, prewhitening the tangent vectors locally converts a Riemannian manifold to an Euclidean space so that the additive parameter update sequence approximates geodesics. Second, we prove that dimensionality reduction of natural gradients is necessary for learning multidimensional linear transformations. Removal of minor components also leads to noise reduction and better computational efficiency. The proposed method demonstrates faster and more robust convergence in the simulations on recovering a Gaussian mixture of artificial data and on discriminative learning of ionosphere data.**

I. Introduction

Optimization based on steepest gradients usually performs poorly in many machine learning problems where the parameter space is not Euclidean. It was pointed out by Amari [1] that the geometry of the Riemannian space must be taken into account when calculating the learning directions. He suggested the use of *natural gradient* updates in place of the ordinary gradient-based ones. Optimization that employs natural gradients generally requires much fewer iterations than the conventional steepest gradient descent (ascend) method. Recently the natural gradient approach has been widely used in learning tasks such as Blind Source Separation [1], Multilayer Perceptrons [1], [3] and optimization in Stiefel (Grassmann) manifolds [8].

The additive natural gradient update rule mimics its ordinary steepest gradient counterpart. However, such an additive rule with natural gradients has no concrete geometric meaning because the space is not Euclidean any more. The resulting updates may severely deviate from the geodesic which connects to the optimal solution.

When natural gradients are applied to *Information Geometry* [2], the Riemannian metric tensor is commonly defined by the Fisher information matrix. Calculating this matrix requires expectation of the tensor product of the score vector, which is usually replaced by the sample average in practice. The optimization based on natural gradients then becomes locally equivalent to the Gauss-Newton method [1], where the Moore-Penrose pseudo inverse is used if the approximated Hessian matrix is singular. For learning perceptrons under the additive Gaussian noise assumption, the computation of natural gradients can be further simplified without explicit matrix inversion (see e.g. [1], [3]).

We propose here to improve the optimization in Information Geometry by using *Principal Component Analysis* (PCA). The gradient vector is whitened with respect to the Fisher information, which results in better approximations to the geodesic updates. The whitening procedure is accompanied with dimensionality reduction for computational efficiency and denoising. We also prove that reducing the dimensions is necessary for learning multidimensional linear transformations. The new method is called *Principal Natural Gradients* (PNAT) after PCA.

We demonstrate the advantages of PNAT by two simulations. The first task is to recover the component means of a Gaussian mixture model by the maximum likelihood method. The second is to learn a matrix that maximizes discrimination of labeled ionosphere data. In both simulations, the geodesic updates with principal natural gradients outperform the original natural gradient updates in efficiency and robustness.

The remaining of this paper is organized as follows. First the basics of natural gradient updates and Information Geometry are briefly reviewed in Section II. In Section III we discuss the motivations of using PCA and present our new method. Next we demonstrate the simulation results on learning a Gaussian mixture model and discriminating ionosphere data in Section IV. Finally the conclusions are drawn in Section V.

II. Natural Gradient Updates

A Riemannian metric is a generalization of the Euclidean one. In an m-dimensional Riemannian manifold, the inner product of two tangent vectors $\mathbf{u}$ and $\mathbf{v}$ at point $\boldsymbol{\theta}$, is defined as

$$\langle \mathbf{u}, \mathbf{v} \rangle_{\boldsymbol{\theta}} \equiv \sum_{i=1}^{m} \sum_{j=1}^{m} G_{ij}(\boldsymbol{\theta}) u_i v_j = \mathbf{u}^T \mathbf{G}(\boldsymbol{\theta}) \mathbf{v}, \qquad (1)$$

where $\mathbf{G}(\boldsymbol{\theta})$ is a positive definite matrix. A Riemannian metric reduces to Euclidean when $\mathbf{G}(\boldsymbol{\theta}) = \mathbf{I}$, the identity matrix.

The steepest descent direction of a function $\mathcal{J}(\boldsymbol{\theta})$ in a Riemannian manifold is represented by the vector $\mathbf{a}$ that minimizes

$$\mathcal{J}(\boldsymbol{\theta}) + \nabla \mathcal{J}(\boldsymbol{\theta})^T \mathbf{a} \qquad (2)$$

under the constraint $\langle \mathbf{a}, \mathbf{a} \rangle_{\boldsymbol{\theta}} = \epsilon$ where ϵ is an infinitesimal constant [1]. By introducing a Lagrange multiplier λ and setting the gradient with respect to $\mathbf{a}$ to zero, one obtains

$$\mathbf{a} = \frac{1}{2\lambda} \mathbf{G}(\boldsymbol{\theta})^{-1} \nabla \mathcal{J}(\boldsymbol{\theta}). \qquad (3)$$

Observing $\mathbf{a}$ is proportional to $\mathbf{G}(\boldsymbol{\theta})^{-1} \nabla \mathcal{J}(\boldsymbol{\theta})$, Amari [1] defined the natural gradient

$$\tilde{\nabla} \mathcal{J}(\boldsymbol{\theta}) \equiv \mathbf{G}(\boldsymbol{\theta})^{-1} \nabla \mathcal{J}(\boldsymbol{\theta}) \qquad (4)$$

and suggested the update rule

$$\boldsymbol{\theta}^{\text{new}} = \boldsymbol{\theta} - \eta \mathbf{G}(\boldsymbol{\theta})^{-1} \nabla \mathcal{J}(\boldsymbol{\theta}) \qquad (5)$$

Zhirong Yang and Jorma Laaksonen are with Laboratory of Computer and Information Science, Helsinki University of Technology, P.O. Box 5400, FI-02015 TKK, Espoo, Finland, email {zhirong.yang, jorma.laaksonen}@tkk.fi

with η a positive learning rate.

Many statistical learning problems can be reduced to probability density estimation. In information theory, the information difference between the current estimated distribution $p \equiv p(\mathbf{x};\boldsymbol{\theta})$ and the optimal one $p^* \equiv p(\mathbf{x};\boldsymbol{\theta}^*)$ can be measured locally by the Kullback-Leibler divergence:

$$D_{\mathrm{KL}}(p,p^*) \equiv \int p(\mathbf{x};\boldsymbol{\theta}^*)\log\frac{p(\mathbf{x};\boldsymbol{\theta}^*)}{p(\mathbf{x};\boldsymbol{\theta})}d\mathbf{x}. \tag{6}$$

Under the regularity conditions

$$\int \frac{\partial}{\partial\boldsymbol{\theta}}p(\mathbf{x};\boldsymbol{\theta})d\mathbf{x} = 0 \tag{7}$$

and

$$\int \frac{\partial^2}{\partial\boldsymbol{\theta}^2}p(\mathbf{x};\boldsymbol{\theta})d\mathbf{x} = 0 \tag{8}$$

the second order Taylor expansion of (6) at p^* is reduced to

$$D_{\mathrm{KL}}(p,p*) \approx \frac{1}{2}\langle d\boldsymbol{\theta}, d\boldsymbol{\theta}\rangle_{\boldsymbol{\theta}^*} = \frac{1}{2}d\boldsymbol{\theta}^T\mathbf{G}(\boldsymbol{\theta}^*)d\boldsymbol{\theta}, \tag{9}$$

where $d\boldsymbol{\theta} = \boldsymbol{\theta} - \boldsymbol{\theta}^*$. The constant $1/2$ has no effect in optimization and is usually omitted. Here the Riemannian structure of the parameter space of a statistical model is defined by the Fisher information matrix [11]

$$G_{ij}(\boldsymbol{\theta}) = E\left\{\frac{\partial\ell(\mathbf{x};\boldsymbol{\theta})}{\partial\theta_i}\frac{\partial\ell(\mathbf{x};\boldsymbol{\theta})}{\partial\theta_j}\right\} \tag{10}$$

in the component form, where $\ell(\mathbf{x};\boldsymbol{\theta}) \equiv -\log p(\mathbf{x};\boldsymbol{\theta})$.

Statistical learning based on (10) was proposed in Amari's Information Geometry theory [2]. The associated natural gradient learning methods have been applied to, for example, Blind Source Separation [1] and Multilayer Perceptrons [1], [3]. However, it is worth to notice that a small η in (5) may not guarantee the learning efficiency of natural gradient updates because it is still unknown how to determine the optimal λ in (3). Natural gradient updates work correctly only when the learning rates at each iteration are properly chosen [1], [13].

III. The Proposed PNAT Method

A. Geodesic Updates

Geodesics in a Riemannian manifold generalize the concept of line segments in an Euclidean space. Given $t \in \mathbb{R}$, a curve $\boldsymbol{\theta} \equiv \boldsymbol{\theta}(t)$ is a *geodesic* if and only if [10]

$$\frac{d^2\theta_k}{dt^2} + \sum_{i=1}^{m}\sum_{j=1}^{m}\frac{d\theta_j}{dt}\frac{d\theta_i}{dt}\Gamma_{ij}^k = 0,\ \forall\ k = \{1,\ldots,m\}, \tag{11}$$

where

$$\Gamma_{ij}^k \equiv \frac{1}{2}\sum_{l=1}^{m}\left(\mathbf{G}^{-1}\right)_{kl}\left(\frac{\partial G_{il}}{\partial\theta_j} + \frac{\partial G_{lj}}{\partial\theta_i} - \frac{\partial G_{ij}}{\partial\theta_l}\right) \tag{12}$$

are Riemannian connection coefficients. The geodesic with a starting point $\boldsymbol{\theta}(0)$ and a tangent vector $\mathbf{v}$ is denoted by $\boldsymbol{\theta}(t;\boldsymbol{\theta}(0),\mathbf{v})$. The *exponential map* of the starting point is then defined as

$$\exp_{\mathbf{v}}(\boldsymbol{\theta}(0)) \equiv \boldsymbol{\theta}(1;\boldsymbol{\theta}(0),\mathbf{v}). \tag{13}$$

It can be shown that the length along the geodesic between $\boldsymbol{\theta}(0)$ and $\exp_{\mathbf{v}}(\boldsymbol{\theta}(0))$ is $|\mathbf{v}|$ [10].

The above concepts are appealing because a geodesic connects two points in the Riemannian manifold with minimum length. Iterative application of exponential maps therefore forms an approximation of flows along the geodesic and the optimization can converge quickly.

Usually obtaining the exponential map (13) is not a trivial task. When the Riemannian metric is accompanied with left- and right-translation invariance, the exponential maps coincide with the ones used in Lie Group theory and can be accomplished by matrix exponentials of natural gradients (see e.g. [8]). However, this property generally does not hold for Information Geometry. Furthermore, solving the differential equations (11) requires calculating the gradient of the Fisher information matrix in (12), which is generally computationally infeasible. A simplified form of exponential maps without explicit matrix inversion can only be obtained in some special cases, for instance, when learning the multilayer perceptrons based on additive Gaussian noise assumptions [1], [3].

It is worth to notice that additive updates are equivalent to exponential maps when the Riemannian metric becomes Euclidean. The Fisher information matrix is the covariance of $\nabla_{\boldsymbol{\theta}}\ell(\mathbf{x};\boldsymbol{\theta})$ and hence positive semi-definite. One can always decompose such a matrix as $\mathbf{G} = \mathbf{EDE}^T$ by singular value decompositions. Denote the whitening matrix as

$$\mathbf{G}^{-\frac{1}{2}} = \mathbf{E}\mathbf{D}^{-\frac{1}{2}}\mathbf{E}^T, \tag{14}$$

where

$$\left(\mathbf{D}^{-\frac{1}{2}}\right)_{ii} = \begin{cases} 1/\sqrt{D_{ii}} & \text{if } D_{ii} > 0 \\ 0 & \text{if } D_{ii} = 0. \end{cases} \tag{15}$$

If the tangent vectors $\mathbf{u}$ and $\mathbf{v}$ are whitened by $\mathbf{G}^{-\frac{1}{2}}$:

$$\tilde{\mathbf{u}} = \mathbf{G}^{-\frac{1}{2}}\mathbf{u},\ \tilde{\mathbf{v}} = \mathbf{G}^{-\frac{1}{2}}\mathbf{v}, \tag{16}$$

the inner product of $\mathbf{u}$ and $\mathbf{v}$ in the Riemannian space becomes $\mathbf{u}^T\mathbf{G}\mathbf{v} = \tilde{\mathbf{u}}^T\tilde{\mathbf{v}}$. That is, the whitened tangent space becomes Euclidean. The additive update rule

$$\boldsymbol{\theta}^{\text{new}} = \boldsymbol{\theta} - \eta\mathbf{G}^{-\frac{1}{2}}\nabla_{\boldsymbol{\theta}}\ell(\mathbf{x};\boldsymbol{\theta}), \tag{17}$$

using the whitened gradient thus forms an exponential map in the local Euclidean space. Here the ordinary online gradient $\nabla_{\boldsymbol{\theta}}\ell(\mathbf{x};\boldsymbol{\theta})$ can be replaced by the batch gradient $\nabla_{\boldsymbol{\theta}}\mathcal{J}(\boldsymbol{\theta})$.

The square root operation on the eigenvalues of the Fisher information matrix distinguishes the proposed method from the original natural gradient updates (5). By contrast, the latter do not form exponential maps unless the metric is Euclidean and the updating sequence may therefore be far from the true geodesic.

B. Principal Components of Natural Gradients

Singular value decomposition used in (14) is tightly connected to *Principal Component Analysis* (PCA) which is usually accompanied with dimensionality reduction for the following two reasons.

First, the Fisher information matrix is commonly estimated by the scatter matrix with $n < \infty$ samples $\mathbf{x}^{(i)}$:

$$\mathbf{G} \approx \frac{1}{n} \sum_{i=1}^{n} \nabla_{\boldsymbol{\theta}} \ell(\mathbf{x}^{(i)}; \boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \ell(\mathbf{x}^{(i)}; \boldsymbol{\theta})^T. \tag{18}$$

However, the estimation accuracy could be poor because of sparse data in high-dimensional learning problems. Principal Component Analysis is a known method for reducing the dimensions and for linear denoising (cf. e.g. [12]). For gradient vectors, PCA discards the minor components that probably irrelevant for the true learning direction, which usually leads to more efficient convergence. Moreover, singular value decomposition (14) runs much faster than inverting the whole matrix $\mathbf{G}$ when the number of principal components is far less than the dimensionality of $\boldsymbol{\theta}$ [6]. The optimization efficiency and robustness can therefore be improved by preserving only the principal components of natural gradients.

Second, dimensionality reduction is sometimes motivated by structural reasons. Consider a problem where one tries to optimize a function of the form

$$\mathcal{J}\left(\|\mathbf{W}^T \mathbf{x}\|^2\right), \tag{19}$$

where $\mathbf{W}$ is an $m \times r$ matrix. Many objectives where Gaussian basis functions are used and evaluated in the linear transformed space can be reduced to this form. Examples can be found in Neighborhood Component Analysis [5] and in a variant of Independent Component Analysis [7]. The following theorem justifies the necessity of dimensionality reduction for this kind of problems.

Denote $\xi_i = \|\mathbf{W}^T \mathbf{x}^{(i)}\|^2$ and $f_i = 2\partial \mathcal{J}(\xi_i)/\partial \xi_i$. The gradient of $\mathcal{J}(\mathbf{x}^{(i)}; \mathbf{W})$ with respect to $\mathbf{W}$ is denoted by

$$\nabla^{(i)} \equiv \nabla_{\mathbf{W}} \mathcal{J}(\mathbf{x}^{(i)}; \mathbf{W}) = f_i \mathbf{x}^{(i)} \left(\mathbf{x}^{(i)}\right)^T \mathbf{W}. \tag{20}$$

The $m \times r$ matrix $\nabla^{(i)}$ can be represented as a row vector $\left(\boldsymbol{\psi}^{(i)}\right)^T$ by concatenating the columns such that

$$\psi^{(i)}_{k+(l-1)m} = \nabla^{(i)}_{kl}, \ k = 1, \ldots, m, \ l = 1, \ldots, r. \tag{21}$$

Piling up the rows $\left(\boldsymbol{\psi}^{(i)}\right)^T$, $i = 1, \ldots, n$, yields an $n \times mr$ matrix $\boldsymbol{\Psi}$.

Theorem 1: Suppose $m > r$. For any positive integer n, the column rank of $\boldsymbol{\Psi}$ is at most $mr - r(r-1)/2$.

The proof can be found in the Appendix. In words, no matter how many samples are available, the matrix $\boldsymbol{\Psi}$ is not full rank when $r > 1$, and neither is the approximated Fisher information matrix

$$\mathbf{G} \approx \frac{1}{n} \boldsymbol{\Psi}^T \boldsymbol{\Psi}. \tag{22}$$

That is, $\mathbf{G}$ is always singular for learning multidimensional linear transformations in (19). $\mathbf{G}^{-1}$ hence does not exist and one must employ dimensionality reduction before inverting the matrix.

Suppose $\hat{\mathbf{D}}$ is a diagonal matrix with the largest q eigenvalues of $\mathbf{G}$, and the corresponding eigenvectors form the columns of matrix $\hat{\mathbf{E}}$. The geodesic update rule with principal components of natural gradient then becomes

$$\boldsymbol{\theta}^{\text{new}} = \boldsymbol{\theta} + \eta \hat{\mathbf{G}}^{-\frac{1}{2}} \boldsymbol{\psi}^{(i)}, \tag{23}$$

where $\hat{\mathbf{G}} = \hat{\mathbf{E}} \hat{\mathbf{D}} \hat{\mathbf{E}}^T$. For learning problems (19), the new $\mathbf{W}$ is then obtained by reshaping $\boldsymbol{\theta}$ into an $m \times r$ matrix. Similar to (17), $\boldsymbol{\psi}^{(i)}$ in (23) can be replaced by a batch gradient. We call the new method *Principal Natural Gradient* (PNAT).

IV. Simulations

A. Gaussian Mixtures

We first tested the PNAT method (23) and the original natural gradient rule (5) on synthetic data. The training samples were generated by a *Gaussian Mixture Model* (GMM) of ten one-dimensional distributions $\mathcal{N}(\mu_i, 1)$, with μ_i randomly chosen within (0,1). We drew 100 scalars from each Gaussian and obtained 1,000 samples in total. Suppose the true μ_i values are unknown to the compared learning methods. The learning task is to recover these means with $\boldsymbol{\mu}$ randomly initialized. We used the maximum likelihood objective, or equivalently the minimum negative log-likelihood

$$\begin{aligned} \mathcal{J}_{\text{GMM}}(\boldsymbol{\mu}) &\equiv \sum_{i=1}^{1000} \ell(x^{(i)}; \boldsymbol{\mu}) \\ &= -\sum_{i=1}^{1000} \log \sum_{j=1}^{10} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^2\right) + C, \end{aligned} \tag{24}$$

where $C = 1000 \log (2\pi)^{10/2}$ is a constant. We then computed the partial derivatives

$$\frac{\partial \ell(x^{(i)}; \boldsymbol{\mu})}{\partial \mu_k} = -\sum_{i=1}^{1000} \frac{(x^{(i)} - \mu_k) \exp\left(-\frac{1}{2}(x^{(i)} - \mu_k)^2\right)}{\sum_{j=1}^{10} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^2\right)}, \tag{25}$$

and approximated the Fisher information matrix by (18).

Figure 1 shows the evolution of the objective when using natural gradients (5) with the learning rate set to $\eta = 10^{-7}$. In the beginning $\mathcal{J}_{\text{GMM}}$ seems to decrease efficiently, but after two iterations this trend ceases and the change becomes slow. At the 54th iteration, we caught a computation warning that the Fisher information matrix is close to singular and its inverse may be inaccurate. Consequently the objective increases dramatically to a value which is even worse than the initial one. At the 56th iteration, the matrix becomes totally singular to the working precision, after which the learning stagnates on a useless plateau.

One may think that the $\eta = 10^{-7}$ could be too large for natural gradient updates and turn to a smaller one, e.g. $\eta = 10^{-8}$. The training result is shown in Figure 2, from which we can see that the objective decreases gradually but slowly. In addition, the evolution curve is not smooth. At each jump we obtained a matrix-close-to-singular warning.

The objectives by using our PNAT method is also shown in Figure 2 (dash-dotted curve). Here we set $\eta = 10^{-3}$ and three principal components of the natural gradient are used in (23). It can be seen that $\mathcal{J}_{\text{GMM}}$ decreases steadily and

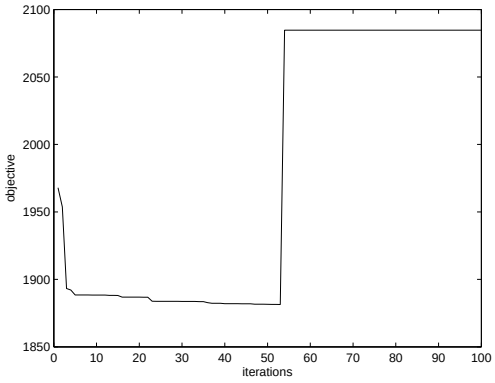

Fig. 1. Learning GMM model by natural gradients with $\eta = 10^{-7}$.

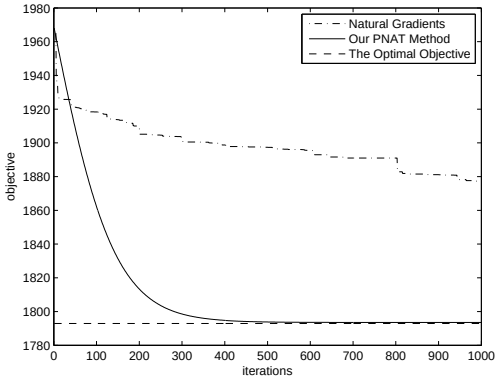

Fig. 2. Learning GMM model by natural gradients with $\eta = 10^{-8}$ and by the PNAT method with $\eta = 10^{-3}$.

efficiently. Within 300 iterations, the loss becomes less than 1800, which is much better than those obtained by original natural gradient updates. The final objective achieved by PNAT update is 1793.53—a value very close to the global optimum 1792.95 computed by using the true Gaussian means.

B. Ionosphere Data

Next, we applied the compared methods on the real ionosphere data set which is available at [4]. The ionosphere data consists of $n = 351$ instances, each of which has 33 real numeric attributes. 225 samples are labeled *good* and the other 126 as *bad*.

Denote $\{(\mathbf{x}^{(i)}, c_i)\}$, $\mathbf{x}^{(i)} \in \mathbb{R}^{33}$, $c_i \in \{\text{good,bad}\}$ for the labeled data pairs. The *Informative Discriminant Analysis* (IDA) [9] seeks a linear transformation matrix $\mathbf{W}$ of size 33×2 such that the negative discrimination

$$\mathcal{J}_{\text{IDA}}(\mathbf{W}) \equiv -\sum_{i=1}^{n} \log p(c_i|\mathbf{y}_i) \tag{26}$$

is minimized in the transformed space where $\mathbf{y}_i = \mathbf{W}^T\mathbf{x}^{(i)}$. The IDA method is aimed at discriminative feature extraction and visualization. The IDA generative model was first discussed in [9], where the predictive density $p(c_i|\mathbf{y}_i)$ is estimated by the Parzen window method:

$$p(c_i|\mathbf{y}_i) \propto \frac{\sum_{j=1}^{n} \phi_{ij} e_{ij}}{\sum_{j=1}^{n} e_{ij}}, \tag{27}$$

where

$$e_{ij} \equiv \begin{cases} \exp\left(-\dfrac{\|\mathbf{y}_i - \mathbf{y}_j\|^2}{2\sigma^2}\right) & i \neq j \\ 0 & i = j \end{cases}, \tag{28}$$

and $\phi_{ij} = 1$ if $c_i = c_j$ and 0 otherwise. The IDA learning requires additional steps to select a proper Parzen window width σ, and the transformation matrix $\mathbf{W}$ is constrained to be orthonormal.

PNAT is suitable for the IDA learning since its objective has the form (19). In this illustrative example we aim at demonstrating the advantage of PNAT over natural gradient updates in convergence. We hence remove the orthonormality constraint for simplicity. By this relaxation, the selection of the Parzen window width is absorbed into the learning of the transformation matrix. Denote $\tilde{\mathbf{y}}_i = \mathbf{A}^T\mathbf{x}^{(i)}$ the relaxed transformation with $\mathbf{A}$ an arbitrary $m \times r$ matrix. The modified estimation of the predictive probability then becomes

$$\tilde{p}(c_i|\tilde{\mathbf{y}}_i) \propto \frac{\sum_{j=1}^{n} \phi_{ij} \tilde{e}_{ij}}{\sum_{j=1}^{n} \tilde{e}_{ij}}, \tag{29}$$

where

$$\tilde{e}_{ij} \equiv \begin{cases} \exp\left(-\frac{1}{2}\|\tilde{\mathbf{y}}_i - \tilde{\mathbf{y}}_j\|^2\right) & i \neq j \\ 0 & i = j \end{cases}. \tag{30}$$

The *Regularized Informative Discriminant Analysis* (RIDA) objective in our experiment is to minimize

$$\begin{aligned} \mathcal{J}_{\text{RIDA}}(\mathbf{A}) &\equiv \tilde{\mathcal{J}}_{\text{IDA}}(\mathbf{A}) + \gamma \|\mathbf{A}\|^2_{\text{Frobenius}} \\ &= -\sum_{i=1}^{n} \log \tilde{p}(c_i|\tilde{\mathbf{y}}_i) + \gamma \sum_{k=1}^{33} \sum_{l=1}^{2} A_{kl}^2. \end{aligned} \tag{31}$$

Here we attach a regularization term to the relaxed IDA objective in order to avoid overfitting [15]. The tradeoff parameter γ is set to 0.1346 according to cross-validation results. Denote $\ell(\mathbf{x}^{(i)}, c_i; \mathbf{A}) \equiv -\log \tilde{p}(c_i|\tilde{\mathbf{y}}_i)$. The Fisher information matrix $\mathbf{G}$ is approximated by (18) with the individual gradients

$$\nabla_{\mathbf{A}} \ell(\mathbf{x}^{(i)}, c_i; \mathbf{A}) = \frac{\sum_{j=1}^{n} \tilde{e}_{ij} \mathbf{B}^{(ij)}}{\sum_{j=1}^{n} \tilde{e}_{ij}} - \frac{\sum_{j=1}^{n} \phi_{ij} \tilde{e}_{ij} \mathbf{B}^{(ij)}}{\sum_{j=1}^{n} \phi_{ij} \tilde{e}_{ij}}, \tag{32}$$

where $\mathbf{B}^{(ij)} = \left(\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\right)\left(\mathbf{x}^{(i)} - \mathbf{x}^{(j)}\right)^T \mathbf{A}$. We then obtain the PNAT update rule:

$$\mathbf{A}^{\text{new}} = \mathbf{A} - \eta \left(\mathbf{G}^{-\frac{1}{2}} \nabla_{\mathbf{A}} \tilde{\mathcal{J}}_{\text{IDA}}(\mathbf{A}) + \gamma \mathbf{A}\right). \tag{33}$$

For comparison, we also used the following natural gradient update rule for the RIDA learning:

$$\mathbf{A}^{\text{new}} = \mathbf{A} - \eta \left(\mathbf{G}^{-1} \nabla_{\mathbf{A}} \tilde{\mathcal{J}}_{\text{IDA}}(\mathbf{A}) + \gamma \mathbf{A}\right). \tag{34}$$

We tried three different learning rates in the natural gradient algorithm (34). The results are shown in Figure

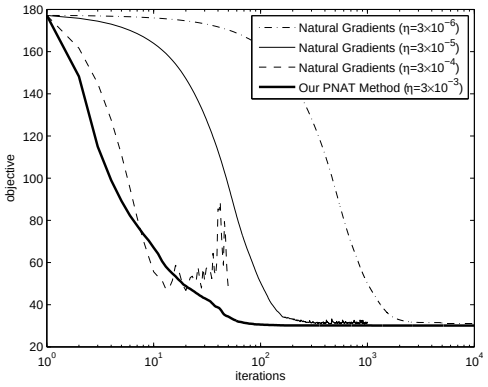

Fig. 3. Objective curves of discriminating ionosphere data by using natural gradient updates and by PNAT.

3. With $\eta = 3 \times 10^{-4}$, the natural gradient updates seem to work well in the first 12 iterations, but after the 13th iteration $\mathcal{J}_{\text{RIDA}}$ jumps from 47.38 to 47.93 and then begins a fluctuation around 50. The same problem happens for $\eta = 3 \times 10^{-5}$ where the curve monotonicity is violated at the 167th iteration. To avoid unexpected jumps one must thus resort to smaller learning rates. By setting $\eta = 3 \times 10^{-6}$, we can see that the objective keeps decreasing in the first 10,000 iterations. However, the learning speed is sacrificed. Natural gradient updates achieve $\mathcal{J}_{\text{RIDA}} < 33$ after 1,944 iterations and the best objective $\mathcal{J}_{\text{RIDA}} = 31.06$ after 10,000 iterations.

By contrast, the proposed PNAT method (33) demonstrates both efficiency and robustness in this learning task. From Figure 3, it can be seen that the negative discrimination keeps decreasing by using PNAT with $\eta = 3 \times 10^{-3}$. We obtained $\mathcal{J}_{\text{RIDA}} < 33$ after 55 iterations and $\mathcal{J}_{\text{RIDA}} < 31.06$ after only 77 iterations. The best objective achieved by using PNAT is 30.12 after 10,000 iterations.

The projected data are displayed in Figure 4, where we used $\eta = 3 \times 10^{-4}$ for natural gradients and $\eta = 3 \times 10^{-3}$ for PNAT. The results are examined after 1,000 iterations of both methods, from which we can see that the *good* samples and *bad* ones are mixed in the middle with the natural gradient updates while for the PNAT case they are well separated. The corresponding objective value is 56.27 by using natural gradients and 30.12 by using PNAT.

It is also interesting to inspect the numbers of principal components used by the PNAT updates (33). We kept the principal components of natural gradients such that the square root sum of the preserved eigenvalues exceeds 95% of the total. The numbers of components in the first 200 iterations are plotted in Figure 5 (top). First we can see that all the numbers are far less than 66, the dimensionality of the gradients while preserving most of the direction information. This verifies the existence of principal variance of gradient vectors. Next, the learning requires more components at the beginning when the model parameters are random and less components in the subsequent iterations. Finally the number becomes stable between 24 and 25. The zoomed

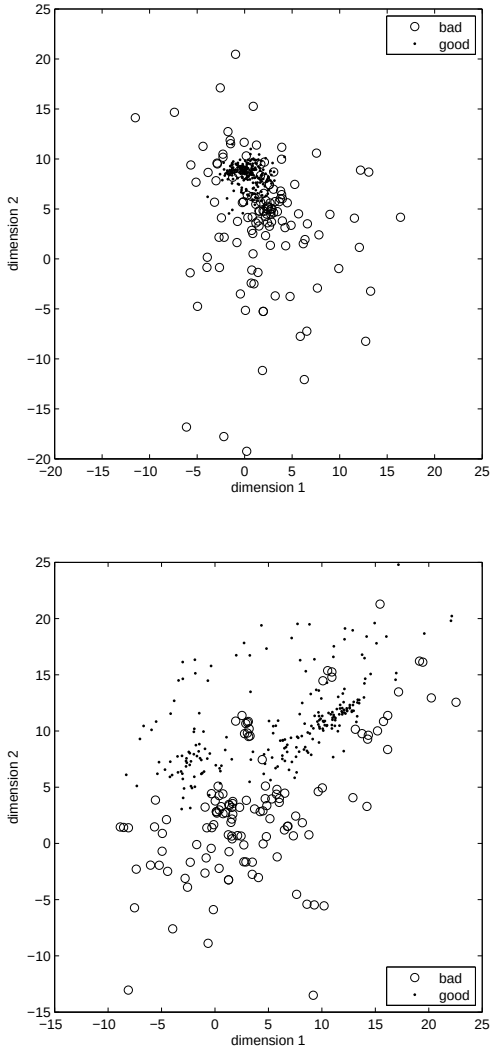

Fig. 4. Ionosphere data in the projected space. Top: 1,000 natural gradient updates with $\eta = 3 \times 10^{-4}$. Bottom: 1,000 PNAT updates with $\eta = 3 \times 10^{-3}$.

objective curve in the fist 200 iterations is also displayed in Figure 5 (bottom) for alignment. It is easy to see that the changing trend of the number of principal components is well consistent with the objective convergence.

V. Conclusions

We have presented a new technique to improve natural gradient updates for optimizations in Information Geometry. Whitening the gradients with respect to the Fisher information matrix transforms the space to be locally Euclidean. As a consequence, the additive updates well approximate the sequence along the geodesic towards the optimal solution. Calculating the learning direction with only principal components of the natural gradients thus enhances both efficiency and robustness. We have also pointed out that dimensionality reduction is indispensable for learning multidimensional linear transformations. The proposed method

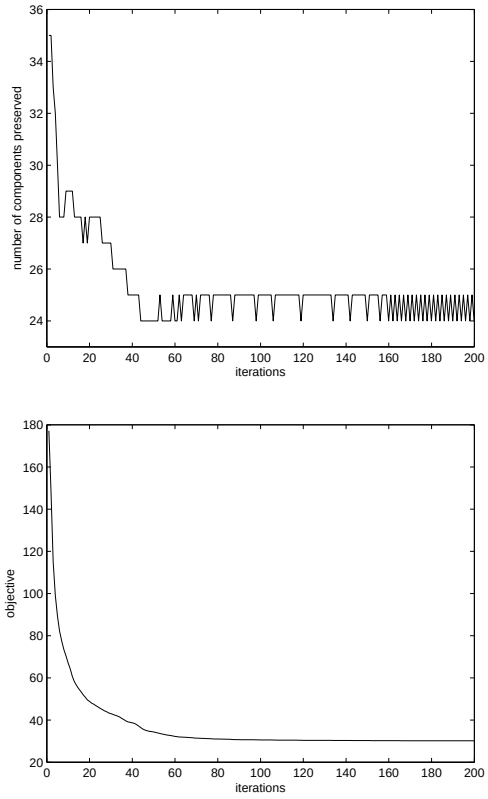

Fig. 5. Top: The numbers of principal components used in the first 200 iterations. Bottom: The aligned objective curve.

has been validated by two simulations, one for recovering a Gaussian mixture model on artificial data and the other for learning a discriminative linear transformation on real ionosphere data.

The proposed approach can potentially be applied to higher-dimensional data such as images or text. Furthermore, adaptive learning in dynamic context may be achieved by adopting online Principal Component Analysis [14].

VI. Acknowledgement

This work is supported by the Academy of Finland in the projects *Neural methods in information retrieval based on automatic content analysis and relevance feedback* and *Finnish Centre of Excellence in Adaptive Informatics Research*.

Appendix

We apply the induction method on r to prove Theorem 1. When $r = 1$, the number of columns of $\mathbf{\Psi}$ is m. Therefore the column rank of $\mathbf{\Psi}$, $\text{rank}_{\text{col}}(\mathbf{\Psi})$, is obvious no greater than $m \times r - r \times (r-1)/2 = m$.

Suppose $\text{rank}_{\text{col}}(\mathbf{\Psi}^{(k-1)}) \leq m(k-1) - (k-1)(k-2)/2$ holds for $r = k-1$, $k \in \mathbb{Z}^+$. Denote $\mathbf{w}^{(j)}$ the j-th column of $\mathbf{W}$. Then we can write $\mathbf{\Psi}^{(k-1)} = \left(\mathbf{B}^{(1)}, \ldots, \mathbf{B}^{(k-1)}\right)$ with block matrix representation $\mathbf{B}^{(j)}$ which equals

$$\begin{pmatrix} f_1 x_1^{(1)} \sum_{d=1}^m x_d^{(1)} w_d^{(j)} & \ldots & f_1 x_m^{(1)} \sum_{d=1}^m x_d^{(1)} w_d^{(j)} \\ \vdots & \ddots & \vdots \\ f_n x_1^{(n)} \sum_{d=1}^m x_d^{(n)} w_d^{(j)} & \ldots & f_n x_m^{(n)} \sum_{d=1}^m x_d^{(n)} w_d^{(j)} \end{pmatrix}. \tag{35}$$

Now consider each of the matrices

$$\tilde{\mathbf{B}}^{(jk)} \equiv \left(\mathbf{B}^{(j)} \;\; \mathbf{B}^{(k)}\right), \; j = 1, \ldots, k-1. \tag{36}$$

Notice that the coefficients

$$\boldsymbol{\rho} \equiv \left(w_1^{(k)}, \ldots, w_m^{(k)}, -w_1^{(j)}, \ldots, -w_m^{(j)}\right) \tag{37}$$

fulfill

$$\boldsymbol{\rho}^T \tilde{\mathbf{B}}^{(jk)} = 0, \; j = 1, \ldots, k-1. \tag{38}$$

Treating the columns as symbolic objects, one can solve the $k-1$ equations (38) by for example Gaussian elimination and then write out the last $k-1$ columns of $\mathbf{\Psi}^{(k)}$ as linear combinations of the first $mr - (k-1)$ columns. That is, at most $m - (k-1)$ linearly independent dimensions can be added when $\mathbf{w}^{(k)}$ is appended. The resulting column rank of $\mathbf{\Psi}^{(k)}$ is therefore no greater than

$$m(k-1) - \frac{(k-1)(k-2)}{2} + m - (k-1) = mk - \frac{k(k-1)}{2}. \tag{39}$$

□

References

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

[2] S. Amari and H. Nagaoka. *Methods of information geometry*. Oxford University Press, 2000.

[3] S. Amari, H. Park, and K. Fukumizu. Adaptive method of realizing natural gradient learning for multilayer perceptrons. *Neural Computation*, 12(6):1399–1409, 2000.

[4] C.L. Blake D.J. Newman, S. Hettich and C.J. Merz. UCI repository of machine learning databases `http://www.ics.uci.edu/~mlearn/MLRepository.html`, 1998.

[5] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. *Advances in Neural Information Processing*, 17:513–520, 2005.

[6] Gene H. Golub and Charles F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, 2 edition, 1989.

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

[8] Y. Nishimori and S. Akaho. Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold. *Neurocomputing*, 67:106–135, 2005.

[9] J. Peltonen and S. Kaski. Discriminative components of data. *IEEE Transactions on Neural Networks*, 16(1):68–83, 2005.

[10] P. Peterson. *Riemannian geometry*. Springer, New York, 1998.

[11] C. R. Rao. Information and accuracy attainable in the estimation of statistical parameters. *Bulletin of the Calcutta Mathematical Society*, 37:81–91, 1945.

[12] J. Särelä and H. Valpola. Denoising source separation. *Journal of Machine Learning Research*, 6:233–272, 2005.

[13] W. Wan. Implementing online natural gradient learning: problems and solutions. *IEEE Transactions on Neural Networks*, 17(2):317–329, 2006.

[14] B. Yang. Projection approximation subspace tracking. *IEEE Transactions on Signal Processing*, 43(1):95–107, 1995.

[15] Zhirong Yang and Jorma Laaksonen. Regularized neighborhood component analysis. In *Proceedings of 15th Scandinavian Conference on Image Analysis (SCIA)*, pages 253–262, Aalborg, Denmark, 2007.