Letters

Performance of the Bayesian Online Algorithm for the Perceptron

Evaldo Araújo de Oliveira and Roberto Castro Alamino

***Abstract*—In this letter, we derive continuum equations for the generalization error of the Bayesian online algorithm (BOnA) for the one-layer perceptron with a spherical covariance matrix using the Rosenblatt potential and show, by numerical calculations, that the asymptotic performance of the algorithm is the same as the one for the optimal algorithm found by means of variational methods with the added advantage that the BOnA does not use any inaccessible information during learning.**

***Index Terms*—Bayesian algorithms, online gradient methods, pattern classification.**

I. Introduction

Online algorithms have great importance in applications mainly because, if suitably designed, they can be able to adapt to situations where the rule is changing although, in general, they perform worse than offline algorithms in static scenarios.

The optimal performance of any perceptron learning rule is achieved by the so-called *Bayes learning rule* which gives rise to a lower bound for the generalization error that cannot be surpassed by any other learning algorithm [4]. It is also generally accepted that online Bayesian methods should perform better than non-Bayesian ones because the former use the available information in the best possible way.

Based on this and in the positive results obtained by the application of the Bayesian approach to a broad range of different situations, a lot of work[1] on Bayesian methods for machine learning has been made. However, exact Bayesian methods turned out to be computationally time-consuming and approximations had to be developed. One important particular approximation, from now on called by us the *Bayesian online algorithm* (BOnA), was proposed and analyzed by Opper [8] for online learning on perceptrons and relies on a projection of the posterior probabilities of the parameters to be estimated on a space of tractable distributions minimizing the Kullback–Leibler divergence between both.

A different approach to learning is provided by variational methods. Variational methods rely on minimizing the generalization error in each step of learning to obtain the best possible performance in each case. Applying a variational method to a one-layer perceptron learning with a Hebbian rule, Kinouchi and Caticha [5] were able to show by means of numerical calculations that the asymptotic behavior of its generalization error when $\alpha \to \infty$, where α is a scaling parameter proportional to the number of examples, is approximately $0.88/\alpha$, which turns out to be two times that of the offline Bayes learning rule. However, the derived algorithm makes use of an unaccessible information: the teacher field (to be defined later). This problem is circumvented in the cited paper by using the mean of this variable as an estimator of its true value.

In this letter, we derive continuum equations for the generalization error of the one-layer perceptron learning by the BOnA with a simplified covariance matrix, which we assume to be spherical, and compare the resulting generalization curve with the optimal algorithm obtained using the variational method in [5]. We show that the performance of the Bayesian algorithm coincides with the performance of the optimal algorithm with the additional advantage that there is no need to use any unaccessible parameter, just the information available in the given data set.

The rest of this letter is organized as follows. In Section II, we review the variational approach to online learning given in [5]. In Section III, the Bayesian method is presented and the Bayesian online algorithm is described. In Section IV, we write the Bayesian simplified equations and finally, in Section V, we discuss the results.

II. Variational Algorithm

Let us consider the supervised learning situation where a one-layer perceptron with N input units and parameterized by its synaptic weights $\omega \in \mathbb{R}^N$ is trained with a data set of examples given by pairs $y_\mu = (\xi_\mu, \sigma_\mu)$, where $\sigma_\mu \in \{-1, 1\}$ is the answer given by a teacher perceptron with synaptic weights $\omega^* \in \mathbb{R}^N$ to the input vector ξ_μ. The teacher is normalized as $\|\omega^*\| = 1$.

A variational algorithm for a one-layer perceptron learning by a Hebbian rule is given in [5]. Using the update equation given by

$$\omega_{\mu+1} = \omega_\mu + \frac{1}{N} W_\mu \sigma_\mu \xi_\mu \tag{1}$$

the modulation function that gives the best gain in generalization ability per example is found by taking the functional derivative with respect to W_μ of the variation rate of ρ, the overlap of synaptic vectors of the teacher and the student, with the number of examples and equating it to zero. The solution is given by

$$W_\mu^* = \|\omega\| \left(\frac{\sigma_\mu b_\mu}{\rho} - \sigma_\mu h_\mu \right) \tag{2}$$

where $b_\mu = \omega^* \cdot \xi_\mu$ and $h_\mu = \omega_\mu \cdot \xi_\mu / \|\omega_\mu\|$ are known, respectively, as the teacher and student fields.

However, the above modulation function depends on a variable which is not accessible in most practical applications: the teacher field b_μ. In the cited paper, the authors use an estimative for W given by its expected value over $|b|$

$$\hat{W}_\mu = \frac{\int d|b| P(b,h) W_\mu^*}{\int d|b| P(b,h)}. \tag{3}$$

The asymptotic behavior of the resulting algorithm for $\alpha \to \infty$, $\alpha = P/N$, where P is the number of examples, is shown to be approximately $0.88/\alpha$ by numerical calculations (assuming a spherical distribution for ξ). This implies that the performance for a large number of examples of this algorithm is approximately two times worse than that of the offline Bayesian algorithm [4].

Manuscript received January 4, 2006; revised October 23, 2006; accepted November 2, 2006. The work of E. de Oliveira was supported in part by Fundação de Apoio à Pesquisa do Estado de São Paulo (FAPESP) under Grant 05/60141-0. The work of R. C. Alamino was supported by the Evergrow Project.

E. A. de Oliveira is with the Instituto de Astronomia, Geofísica e Ciências Atmosféricas, Universidade de São Paulo, São Paulo, CEP 01060-970, Brazil (e-mail: evaldo@model.iag.usp.br).

R. C. Alamino is with the Neural Computing Research Group, Aston University, Birmingham B4 7ET, U.K. (e-mail: alaminrc@aston.ac.uk; robertoalamino@yahoo.com).

Digital Object Identifier 10.1109/TNN.2007.891189

[1] This can be seen by the crescent amount of papers on Bayesian methods presented at the Neural Information Processing Systems (NIPS) Conference—http://www.nips.cc/.

III. BOnA

The BOnA was proposed by Opper and studied in some detail in [8]. Consider the general case where a set of parameters $\omega \in \mathbb{R}^N$ needs to be estimated for some model based on a set D_P of P examples y_μ, $\mu = 1, \ldots, P$. Before the beginning of the training procedure, an *a priori* parametric distribution $P(\omega|\hat{\omega}_0, C_0)$ of the parameters to be estimated is chosen as a Gaussian with mean and covariance matrix, respectively, given by $\hat{\omega}_0$ and C_0. When a new example is presented, the distribution is updated using the Bayes theorem. This update, however, can take the posterior distribution out from the manifold of Gaussian distributions. The posterior is then projected back into a Gaussian by minimizing the Kullback–Leibler divergence between both distributions (the posterior and the projected). The process is repeated iteratively and, at each iteration, the corresponding estimative for the parameters ω is given by the mean of the Gaussian with its covariance matrix giving a measure of the uncertainty of the estimative. The update equations for both parameters are given, in matrix form, by

$$\hat{\omega}_{\mu+1} = \hat{\omega}_\mu + C_\mu \frac{\partial}{\partial \hat{\omega}_\mu} \ln \langle P(y_{\mu+1}|u + \hat{\omega}_\mu)\rangle_u \tag{4}$$

$$C_{\mu+1} = C_\mu + C_\mu \left(\frac{\partial^2}{\partial \hat{\omega}_\mu^2} \ln \langle P(y_{\mu+1}|u + \hat{\omega}_\mu)\rangle_u \right) C_\mu \tag{5}$$

where $\langle \ldots \rangle_u$ means the average over the zero-mean unit-covariance Gaussian distributed variable $u \in \mathbb{R}^N$ and we used the conventions

$$\left(\frac{\partial f}{\partial \hat{\omega}}\right)_i \equiv \left(\frac{\partial f}{\partial \hat{\omega}_i}\right) \quad \left(\frac{\partial^2 f}{\partial \hat{\omega}^2}\right)_{ij} \equiv \left(\frac{\partial^2 f}{\partial \hat{\omega}_i \hat{\omega}_j}\right) \tag{6}$$

with f an arbitrary function.

IV. BOnA Equations for the Perceptron

Let us apply the BOnA to a perceptron learning situation. For simplicity, we choose the parametric family of distributions to be the spherical Gaussian

$$\mathcal{G}_\mu(\omega) = \exp\left\{-\|\omega - \hat{\omega}_\mu\|^2/2\zeta_\mu\right\} / \sqrt{2\pi\zeta_\mu} \tag{7}$$

with $\hat{\omega}_\mu \in \mathbb{R}^N$ and $\zeta_\mu \in \mathbb{R}_+$. This choice results in an algorithm where the increments to the synaptic weights are made in the direction of the learned example as in a Hebbian rule, defining the likelihood of the parameters using a Rosenblatt potential for the error potential [4], [7] such that

$$P(y|\omega) = \frac{e^{-\beta V(\omega, y)}}{\int d\omega e^{-\beta V(\omega, y)}} \tag{8}$$

with

$$V(\omega, y) = -\sigma \frac{\omega \cdot \xi}{\sqrt{N}} \Theta\left(-\sigma \frac{\omega \cdot \xi}{\sqrt{N}}\right) \tag{9}$$

where $\Theta(x)$ is the Heaviside step function and β a free parameter. In the limit $\beta \to \infty$, the BOnA applied to this particular case is called the *scalar BOnA*. The equations we will obtain are[2]

$$\hat{\omega}_{\mu+1} = \hat{\omega}_\mu + \sigma_{\mu+1}\xi_{\mu+1}\sqrt{\frac{2\zeta_\mu}{\pi N}}\mathcal{F}_\mu \tag{10}$$

$$\zeta_{\mu+1} = \zeta_\mu - \frac{\zeta_\mu \mathcal{F}_\mu}{N}\left(\tau\sqrt{\frac{2}{\pi\zeta_\mu}} + \frac{2}{\pi}\mathcal{F}_\mu\right) \tag{11}$$

with $\tau_\mu = \sigma_{\mu+1}\hat{\omega}_\mu \cdot \xi_{\mu+1}/\sqrt{N}$ and $\mathcal{F}_\mu \equiv \mathcal{F}(-\tau_\mu/\sqrt{2\zeta_\mu})$, where we define the function $\mathcal{F}$ as

$$\mathcal{F}(x) = \frac{e^{-x^2}}{\operatorname{erfc}(x)}. \tag{12}$$

The learning process described by the above equations is a stochastic process, because, in each step, it receives a vector ξ_μ selected randomly from a distribution $P(\xi)$. Therefore, the usual procedure to solve the dynamics would be to calculate the evolution of the probability distributions of the variables we are interested in. However, as we are interested in the behavior in the thermodynamic limit $N \to \infty$, we can write down the differential equations at once and after that calculate the asymptotic behavior of the algorithm.

For the perceptron, we will be interested in the norm of the synaptic vector $\hat{\omega}_\mu$ and its correlation with the teacher vector ω^*, since we are using the generalization error $e_g(\alpha) = (1/\pi)\mathrm{acos}\rho(\alpha)$ as a measure of the performance.

Using rescaled norms $Q_\mu = \hat{\omega}_\mu \cdot \hat{\omega}_\mu/N$ and $M = \omega^* \cdot \omega^*/N$, we have

$$\rho_\mu = \frac{\hat{\omega}_\mu \cdot \omega^*}{N\sqrt{Q_\mu M}}. \tag{13}$$

As the time parameter, we chose $\alpha = \mu/N$ such that $\Delta\alpha = 1/N$, what is extremely convenient for large N and, therefore, for the continuum limit. Thus, starting with (10) and defining $\hat{\tau}_\mu \equiv \tau_\mu/\sqrt{Q_\mu}$, we find

$$Q(\alpha + 1/n) - Q(\alpha) = \frac{2}{N}\left[\sqrt{\frac{2Q(\alpha)}{\pi\zeta(\alpha)^{-1}}}\sigma(\alpha)\hat{\tau}(\alpha) + \frac{\zeta(\alpha)}{\pi}\mathcal{F}\left(-\hat{\tau}(\alpha)\sqrt{\frac{Q(\alpha)}{2\zeta(\alpha)}}\right)\right]\mathcal{F}\left(-\hat{\tau}(\alpha)\sqrt{\frac{Q(\alpha)}{2\zeta(\alpha)}}\right) \tag{14}$$

what gives

$$\begin{aligned} &\frac{Q(\alpha + \nu/N) - Q(\alpha)}{\nu/N} \\ &= \frac{2}{\nu}\sum_{n=0}^{\nu-1}\left[\sqrt{\frac{2Q\left(\alpha + \frac{n}{N}\right)}{\pi\zeta\left(\alpha + \frac{n}{N}\right)^{-1}}} \times \sigma\left(\alpha + \frac{n}{N}\right)\hat{\tau}\left(\alpha + \frac{n}{N}\right) + \frac{\zeta\left(\alpha + \frac{n}{N}\right)}{\pi}\mathcal{F}\left(\frac{-\hat{\tau}\left(\alpha + \frac{n}{N}\right)}{\sqrt{2\zeta\left(\alpha + \frac{n}{N}\right)}}\right)\right] \\ &\times \mathcal{F}\left(\frac{-\hat{\tau}\left(\alpha + \frac{n}{N}\right)}{\sqrt{2\zeta\left(\alpha + \frac{n}{N}\right)}}\right). \end{aligned} \tag{15}$$

In the limit $\nu \to \infty$ and $N \to \infty$, n/N becomes a continuum variable, the left-hand side of (15) becomes a time derivative (with respect to α) and $(1/\nu)\sum_n^{\nu-1}$ becomes an integral over D, or equivalently, over $\{\sigma, \hat{\tau}, \bar{\tau}\}$. So, we finally find[3]

$$\frac{dQ}{d\alpha} = 2\left\langle\left[\sqrt{\frac{2Q\zeta}{\pi}}\sigma\hat{\tau} + \frac{\zeta}{\pi}\mathcal{F}\left(\frac{-\hat{\tau}}{\sqrt{2\zeta}}\right)\right]\mathcal{F}\left(\frac{-\hat{\tau}}{\sqrt{2\zeta}}\right)\right\rangle_{\sigma,\hat{\tau}}. \tag{16}$$

Summing over σ, we finally find the differential equation for $Q(\alpha)$

[2] As we are interested in the asymptotic regime for α with $N \to \infty$, we consider $\xi \cdot \xi/N = 1(\xi_i \sim \mathcal{N}(0, 1))$ for obtaining (10) and (11).

[3] See [10] for a formal demonstration.

$$\frac{dQ}{d\alpha} = \frac{2\zeta}{\pi} \int_{-\infty}^{\infty} d\hat{\tau} \frac{\mathrm{erfc}(-\hat{\tau}/\sqrt{r})}{\mathrm{erfc}(-\hat{\tau}\sqrt{Q/\zeta})} e^{-(1+Q/\zeta)\hat{\tau}^2} \times \left[2\hat{\tau}\sqrt{\frac{Q}{\zeta}} + \frac{e^{-Q\hat{\tau}^2/\zeta}}{\sqrt{\pi}\mathrm{erfc}(-\hat{\tau}\sqrt{Q/\zeta})}\right] \quad (17)$$

with $r = 1/\rho^2 - 1$.

Following the same procedure, we find for $\zeta(\alpha)$

$$\frac{d\zeta}{d\alpha} = -\frac{2\zeta}{\pi} \int_{-\infty}^{\infty} d\hat{\tau} \frac{\mathrm{erfc}(-\hat{\tau}/\sqrt{r})}{\mathrm{erfc}(-\hat{\tau}\sqrt{Q/\zeta})} e^{-(1+Q/\zeta)\hat{\tau}^2} \times \left[\hat{\tau}\sqrt{\frac{Q}{\zeta}} + \frac{e^{-Q\hat{\tau}^2/\zeta}}{\sqrt{\pi}\mathrm{erfc}(-\hat{\tau}\sqrt{Q/\zeta})}\right]. \quad (18)$$

Now, all we need is the equation for ρ. Multiplying (10) by ω^*, we find

$$\rho' = \frac{1}{N\sqrt{Q'}} \left(\omega^* \cdot \hat{\omega} + \sqrt{\frac{2\zeta}{\pi}} \sigma' \bar{\tau}' \mathcal{F}\right) \quad (19)$$

with the variables with $'$ in time $(\alpha + 1/N)$ and the others in α, except ω^* that is kept constant during the learning. In this equation, we defined $\bar{\tau}(\alpha) \equiv \omega^* \cdot \xi(\alpha)/\sqrt{N}$ and $\mathcal{F} \equiv \mathcal{F}(-\hat{\tau}\sqrt{Q/2\zeta})$.

Using (14), we have

$$\frac{1}{\sqrt{Q'}} \simeq \frac{1}{\sqrt{Q}} \left[1 - \left(\sigma\hat{\tau}\sqrt{2Q\zeta/\pi} + \frac{\zeta}{\pi}\mathcal{F}\right) \frac{\mathcal{F}}{NQ}\right] \quad (20)$$

that substituting in (19) leads to

$$\frac{d\rho}{d\alpha} = \left\langle \sqrt{\frac{2\zeta}{\pi Q}} (\bar{\tau} - \rho\hat{\tau})\sigma\mathcal{F} - \frac{\rho\zeta}{\pi Q}\mathcal{F}^2 \right\rangle_{\sigma,\bar{\tau},\hat{\tau}}. \quad (21)$$

Integrating with respect to $\bar{\tau}$ and summing over σ, we get

$$\frac{d\rho}{d\alpha} = \frac{2\rho}{\pi^{3/2}} \int_{-\infty}^{\infty} d\hat{\tau} \frac{e^{-(1+Q/\zeta)\hat{\tau}^2}}{\mathrm{erfc}^2(-\hat{\tau}\sqrt{Q/\zeta})} \times \left[\sqrt{\frac{r\zeta}{Q}} \mathrm{erfc}(-\hat{\tau}\sqrt{Q/\zeta}) e^{-\hat{\tau}^2/r} - \frac{\zeta}{Q}\mathrm{erfc}(-\hat{\tau}/\sqrt{r}) e^{-Q\hat{\tau}^2/\zeta}\right]. \quad (22)$$

In Fig. 1, we show $e_g(\alpha)$ and $\zeta(\alpha)$ for the scalar BOnA, obtained by numerically integrating the coupled differential (17), (18), and (22). As can be seen, the generalization error converges to $0.88/\alpha$, which is exactly the same asymptotic error obtained by the variational algorithm. Nevertheless, the great advantage of this approach is not using inaccessible information like ρ or $\bar{\tau}$. Besides, equating $\eta(\alpha) = \sqrt{\zeta(\alpha)}$ in (10), we see that asymptotically $\eta(\alpha) \propto 1/\alpha$, that for $\eta(\alpha)$ is a sufficient condition for a local convergence of $\hat{\boldsymbol{\omega}}(\alpha)$ to $\bar{\boldsymbol{\omega}}$ [3], [9].

V. Conclusion

One of the main problems of gradient-descent algorithms is the adjustment of the learning rate, which, in order to prevent asymptotic fluctuations, must drop after a transient phase depending on the error potential. A Bayesian approach requires that this transient should be estimated by means of the update of some *a priori* distribution by the algorithm itself. This kind of behavior is clearly noted in the presented algorithms where we have a learning rate proportional to the variance of the *a priori* distribution, e.g., (10) and (11). Although this variance also depends on the error potential, this dependence will not be direct on the given potential V, but on an induced potential $\mathcal{E}_{\mathrm{x}} = -\ln\langle\exp(-\beta V)\rangle$. This change in potentials was observed also in algorithms obtained by variational methods [5], [6].

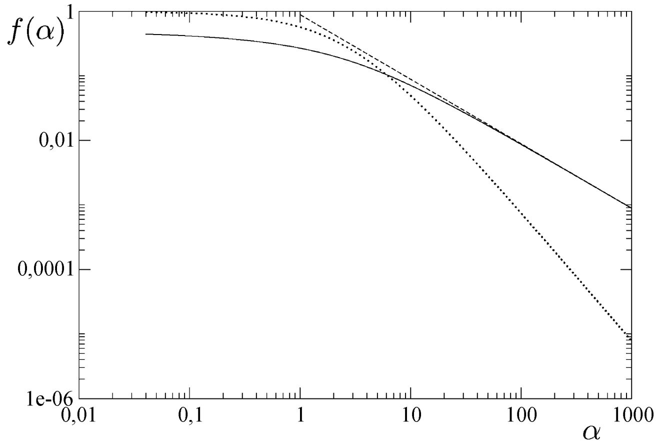

Fig. 1. Numerical solution for the scalar BOnA. The dashed line represents $f(\alpha) = 0.88/\alpha$, the continuous line is $f(\alpha) = e_g(\alpha)$, and the dotted line $f(\alpha) = \zeta(\alpha)$. For large α, we have $e_g(\alpha) \simeq 0.88/\alpha$ and $\zeta \propto \alpha^{-2}$.

TABLE I
ASYMPTOTIC GENERALIZATION ERROR FOR THE ROSENBLATT ONLINE ALGORITHM [4], VARIATIONAL OPTIMAL [5], BOnA, AND THE BAYESIAN RULE (BOffA) [4]

Rosenblatt On.	Variat. Opt.	BOnA	BOffA.
$0.28/\alpha^{1/3}$	$0.88/\alpha$	$0.88/\alpha$	$0.44/\alpha$

We conclude that the BOnA uses the same optimal functional form with respect to the minimization of the generalization error, but the learning rate is differently adjusted. In the optimal algorithm, the learning rate is given by the square root of r, while in BOnA it is given by the *a priori* width. In practical situations, BOnA is the correct choice because we hardly have access to quantities such as ρ, but the key information about the learning rate is revealed in r. In the beginning of the learning process, the correlation between the teacher perceptron and the student is small, which means that r is large. As $\hat{\omega}$ becomes closer to ω^*, $\rho \rightarrow 1$ and $r \rightarrow 0$. In the BOnA, the student has no access to ρ and estimates its correlation with the teacher based on $\langle(\omega - \langle\omega\rangle)^2\rangle_{\omega,D}$.

In Table I, we can see a comparison between the asymptotic behavior of the generalization error in four different algorithms showing that BOnA is as good as the variational algorithm and both are slower than Bayesian offline algorithm (BOffA) only by a factor of two, while the Rosenblatt algorithm has a much slower asymptotic behavior than all the others due two the correspondent exponent of α.

Summarizing, we showed that the BOnA applied to the perceptron with the Rosenblatt potential leads to the same asymptotic performance of the optimal algorithm obtained by variational methods with the advantage that no extra information beyond the data set (e.g., the parameters of the professor) is needed to make the algorithm more adaptive.

Appendix
Obtaining the Update Equations for the BOnA

In order to calculate (4) and (5), we start with the Bayes' rule

$$\hat{\omega}_{\mu+1} = \int d\omega \omega P(\omega|D_{\mu+1}) = \frac{\int d\omega \omega L(y_{\mu+1}|\omega)P(\omega|D_\mu)}{\int d\omega L(y_{\mu+1}|\omega)P(\omega|D_\mu)}$$

where $L(y_{\mu+1}|\omega)$ is the likelihood of the new datum and $P(\omega|D_\mu)$ is the Gaussian distribution

$$P(\omega|D_\mu) = \frac{e^{-(\omega_\mu - \hat{\omega}_\mu)\cdot C_\mu^{-1}(\omega_\mu - \hat{\omega}_\mu)/2}}{\sqrt{(2\pi)^N |C_\mu|}}. \tag{23}$$

This can be written as

$$\hat{\omega}' = \hat{\omega} + \frac{\int du u e^{-\frac{1}{2}u\cdot C^{-1}u} L(y_{\mu+1}|u+\hat{\omega})}{\int du e^{-\frac{1}{2}u\cdot C^{-1}u} L(y_{\mu+1}|u+\hat{\omega})}$$

where we are using $'$ to the time index $\mu + 1$ and we do not use any index when the variables are at time μ (or for integration variables).

Note that $u_i e^{-u\cdot C^{-1}u/2} = -\sum_j C_{ij}\partial_{u_j} e^{-u\cdot C^{-1}u/2}$ with $\partial_{u_j} \equiv (\partial/\partial u_j)$, then one integration by parts leads to

$$\hat{\omega}_i' = \hat{\omega}_i + \sum_j C_{ij} \frac{\int du e^{-\frac{1}{2}u\cdot C^{-1}u} \partial_{u_j} L(y_{\mu+1}|u+\hat{\omega})}{\int du e^{-\frac{1}{2}u\cdot C^{-1}u} L(y_{\mu+1}|u+\hat{\omega})}.$$

Equation (4) is obtained using the identity $\partial_{u_i} F(u_i + \hat{\omega}_i) = \partial_{\hat{\omega}_i} F(u_i + \hat{\omega}_i)$

$$\hat{\omega}_i' = \hat{\omega}_i + \sum_j C_{ij} \partial_{\hat{\omega}_j} \ln \langle L(y_{\mu+1}|u+\hat{\omega})\rangle .$$

For (5), we start defining $\Delta\hat{\omega}_i = \hat{\omega}_i' - \hat{\omega}_i$, then

$$C_{ij} = \int du (u_i - \Delta\hat{\omega}_i)(u_j - \Delta\hat{\omega}_j) P(u + \hat{\omega}|D_{\mu+1}).$$

Now, using the identity

$$u_i u_j = C_{ij} e^{-u\cdot C^{-1}u/2} + \sum_{kl} C_{ik} C_{lj} \partial_{u_k} \partial_{u_l} e^{-u\cdot C^{-1}u/2}$$

and the same tricks used before with two integration by parts, we finally find the update equation for the covariance matrix

$$C_{ij}' = C_{ij} + \sum_{kl} C_{ik} C_{lj} \partial_{\hat{\omega}_k} \partial_{\hat{\omega}_l} \ln \langle L(y_{\mu+1}|u+\hat{\omega})\rangle .$$

Equations (10) and (11) are obtained by following the same way and using (8) and (9) to calculate $\langle P(y_{\mu+1}|u+\hat{\omega}_\mu)\rangle_u$.

Acknowledgment

The authors would like to thank M. Opper and N. Caticha for useful discussions.

References

[1] N. Caticha and E. A. de Oliveira, "Gradient descent learning in and out of equilibrium," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 63, pp. 061905-1–061905-6, 2001.

[2] E. A. de Oliveira, "The Rosenblatt Bayesian algorithm learning in a nonstationary environment," *IEEE Trans. Neural Netw.*, vol. 18, no. 2, Mar. 2007, to be published.

[3] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley, 2001.

[4] A. Engel and C. Van den Broeck, *Statistical Mechanics of Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2001.

[5] O. Kinouchi and N. Caticha, "Optimal generalization in perceptrons," *J. Phys. A. Math. Gen.*, vol. 25, pp. 6243–6250, 1992.

[6] ——, "Learning algorithms that give the Bayes generalization limit for perceptrons," *Phys. Rev. E, Gen. Phys.*, vol. 54, pp. R54–R57, 1996.

[7] E. Levin, N. Tishby, and S. A. Solla, "A statistical approach to learning and generalization in layered neural networks," *Proc. IEEE*, vol. 78, no. 10, pp. 1568–1574, Oct. 1990.

[8] M. Opper, "A Bayesian approach to online learning," in *On-Line Learning in Neural Networks*, D. Saad, Ed. Cambridge, U.K.: Cambridge Univ. Press, 1998, pp. 363–378.

[9] M. Murata, M. Kawanabe, A. Ziehe, K. Müller, and S. Amari, "On-line learning in changing environments with applications in supervised and unsupervised learning," *Neural Netw.*, vol. 15, pp. 743–760, 2002.

[10] G. Reents and R. Urbanczik, "Self-averaging and on-line learning," *Phys. Rev. Lett.*, vol. 80, pp. 5445–5447, 1998.

Variational Bayesian Approach to Canonical Correlation Analysis

Chong Wang

***Abstract*—As a dimension reduction algorithm, canonical correlation analysis (CCA) encounters the issue of selecting the number of canonical correlations. In this letter, we present a Bayesian model selection algorithm for CCA based on a probabilistic interpretation. A hierarchical Bayesian model is applied to probabilistic CCA and learned by variational approximation. This method not only estimates the model parameters, but also automatically determines the number of canonical correlations and avoids overfitting. Experiments show that it performs better compared with maximum likelihood and some other model selection methods.**

***Index Terms*—Bayesian inference, canonical correlation analysis (CCA), dimensionality reduction, model selection, variational approximation.**

I. Introduction

Canonical correlation analysis (CCA) [4], similar to principal component analysis (PCA), is a widely used tool in the dimensionality reduction, feature extraction, and visualization for pattern recognition. CCA and its extended methods have been used in many applications, such as facial expression recognition [2] and text–image modeling [5]. Given two random vectors x_1 and x_2, with dimensions d_1 and d_2, CCA can be used to find the basis vectors for x_1 and x_2, so that the correlation between projections of variables onto these basis vectors is mutually maximized. One of the central problems in CCA is model selection or how to select the dimensions to be retained. In [1], Bach and Jordan propose a novel probabilistic interpretation of CCA with latent variables. With this probabilistic interpretation, various Bayesian treatments can be applied. In this letter, we propose a hierarchical Bayesian model using novel variational approach to address the CCA model selection problem.

Manuscript received April 23, 2006; revised August 1, 2006 and October 17, 2006; accepted November 26, 2006.

The author was with the Department of Automation, Tsinghua University, Beijing 100084, China. He is now with Microsoft Research Asia, Beijing 100080, China (e-mail: wangchong99@mails.tsinghua.edu.cn; chwang@microsoft.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNN.2007.891186