Learning On-Line Classification via Decorrelated LMS Algorithm: Application to Brain-Computer Interfaces

Shiliang Sun and Changshui Zhang

State Key Laboratory of Intelligent Technology and Systems,
Department of Automation, Tsinghua University, Beijing, China, 100084
sunsl02@mails.tsinghua.edu.cn, zcs@mail.tsinghua.edu.cn

Abstract. The classification of time-varying neurophysiological signals, e.g., electroencephalogram (EEG) signals, advances the requirement of adaptability for classifiers. In this paper we address the challenge of neurophysiological signal classification arising from brain-computer interface (BCI) applications and propose an on-line classifier designed via the decorrelated least mean square (LMS) algorithm. Based on a Bayesian classifier with Gaussian mixture models, we derive the general formulation of gradient descent algorithms under the criterion of LMS. Further, to accelerate convergence, the decorrelated gradient instead of the instantaneous gradient is adopted for updating the parameters of the classifier adaptively. Utilizing the presented classifier for the off-line analysis of practical classification tasks in brain-computer interface applications shows its effectiveness and robustness compared to the stochastic gradient descent classifier which uses the instantaneous gradient directly.

1 Introduction

Recently, the emerging research of brain-computer interface (BCI) technology, which is to give its users communication and control routes that do not depend on the brain's normal output channels of peripheral nerves and muscles, issues many challenges to the artificial intelligence community [1][2][3][4]. One of the big challenges in BCI applications is how to recognize the user's intent from the observation of neurophysiological signals as accurate as possible. In this paper, we focus on the classification problem of one particular variety of neurophysiological signals, namely electroencephalogram (EEG) signals which are electrical brain activities recorded from electrodes placed on the scalp.

Compared to magnetoencephalography (MEG), optical imaging, positron emission tomography (PET) and functional magnetic resonance imaging (fMRI), electroencephalography is a relatively inexpensive and convenient means to monitor the brain's activities. Although the recorded EEG signals suffer from the trouble of low signal noise rate (SNR), currently it is a rather recipient way (non-invasive and ethical) to access brain signals [5][6]. However, the essential nondeterminacy of brain activity implies the high variability of EEG recordings.

A. Hoffmann, H. Motoda, and T. Scheffer (Eds.): DS 2005, LNAI 3735, pp. 215–226, 2005.

The EEG signals being used in a BCI are typically non-stationary, especially between two different sessions with a rather long time interval. Factors such as user's strategy, motivation, attention, fatigue or frustration may affect the features of EEG activities significantly. Besides, the environmental noise in all kinds of natural conditions can also cause the mental state to change by gradual degrees. As an instance, Millán had shown that two different mental tasks, imagination of left and right hand movements respectively, can have closer power maps than the same task during two consecutive sessions [7]. Altogether, the spontaneous variability of EEG recordings between experimental sessions makes it a difficult issue to categorize different EEG signals, and necessitates learning the on-line classification to boost up the performance of existing BCIs.

Hitherto, there is few work dealing with the problem of on-line learning for EEG signal classification in the literature. Although many on-line learning methods are available from the neural network, statistical, and computational learning disciplines, they are usually computationally expensive and do not suit BCI applications simply [8][9][10]. Our current work is initially inspired by several recent publications of Millán and his colleagues [7][11][12][13]. Although they presented to use the idea of stochastic gradient descent to carry out on-line learning of a statistical classifier, under their rather rigorous assumptions, they hadn't provided the formulations of variable updates in a systematic way. We will make up for this deficiency and discuss the related work later in the main text.

The main contribution of this article is that, based on a Bayesian classifier with Gaussian mixture models we derive the exact formulation of gradient algorithm in a much general way, and then present a decorrelated least mean square (DLMS) algorithm utilizing the theoretical outcome to learn the on-line classification of EEG signals in BCI applications. Real-world classification experiments with three kind of mental imagery tasks also verifies the effectiveness of our approach.

The remainder of this paper is organized as follows. Besides the theoretical derivation of gradient update, section 2 also covers the details of how to build up the on-line Bayesian classifier employing the idea of decorrelated LMS algorithm. Section 3 reports the experimental results for several BCI subjects on three mental imagery tasks. Then, in section 4 we discuss some related work. Finally, section 5 gives the conclusions and future work plan.

2 On-Line Classifiers

As we have stated before, the competence of on-line learning is very necessary in BCI applications. However, to the best of our knowledge, there is little work addressed this matter in the literature till now. The articles of Millán *et al.* are one of the first to bring forward this problem in the BCI settings [7][11][12][13]. For the on-line learning in BCIs, one would first encounter the problem of choose which kind of classifiers. For the consideration of low computation cost and practical superiority, here we adopt the Bayesian classifier to deal with the issue of multi-class categorization, as suggested by others [7][12].

2.1 Bayesian Classifier

Assume there are N samples in a training set which come from K categories, and each class denoted by C_k has prior $P(C_k)$, $(k = 1, ..., K)$, s.t., $\sum_{k=1}^{K} P(C_k) = 1$. For each class, its class conditional probability density is assumed to be the weighted combination of N_k Gaussian probability density functions, i.e.,

$$p(x|C_k) = \sum_{i=1}^{N_k} a_k^i G(x|\mu_k^i, \Sigma_k^i), s.t., \Sigma_{i=1}^{N_k} a_k^i = 1 \tag{1}$$

where $G(x|\mu_k^i, \Sigma_k^i)$ is a Gaussian probability density function with mean μ_k^i and covariance Σ_k^i [14]. According to Bayesian theorem [10], the posterior probability of x belonging to class C_k can be given as

$$\begin{aligned} P(C_k|x) &= \frac{P(C_k)p(x|C_k)}{p(x)} \\ &= \frac{P(C_k)\sum_{i=1}^{N_k} a_k^i G(x|\mu_k^i, \Sigma_k^i)}{\sum_{j=1}^{K} P(C_j)\sum_{i=1}^{N_j} a_j^i G(x|\mu_j^i, \Sigma_j^i)} . \end{aligned} \tag{2}$$

Now we represent the samples as $\{x_n, y_n\}, n = 1, ..., N$, whereas x_n is the feature vector, y_n is the corresponding label. If $x_n \in C_k$, then $y_n = e_k^K = \left[0, \ldots, 1_{(k)}, \ldots, 0\right]_{(K)}^{\top}$. Denote $\hat{y}_n$ as the outcome of our Bayesian classifier, i.e.,

$$\hat{y}_n = \left[P(C_1|x_n), P(C_2|x_n), \ldots, P(C_K|x_n)\right]^{\top} .$$

Under the criterion of least mean square (LMS), the cost function for unconstrained optimization becomes

$$\min J(\Theta) = \min E\{\|e_n\|^2\} = \min E\{\|y_n - \hat{y}_n\|^2\} \tag{3}$$

where variable Θ represents any of the parameters $N_k, a_k^i, \mu_k^i, \Sigma_k^i$. To make our analysis feasible, we only presume here that parameters N_k, a_k^i are given or obtained from previous training data, while parameters μ_k^i, Σ_k^i would have the most general form (μ_k^i is a general column vector, Σ_k^i is a symmetric and positive definite matrix) and would be updated through on-line learning.

For the application of LMS algorithm and the later mentioned decorrelated LMS algorithm, one should first derive the formulation of stochastic gradient (instantaneous gradient) $\nabla_\Theta \|y_n - \hat{y}_n\|^2$. Note that $\|y_n - \hat{y}_n\|^2$ can be rewritten as follows:

$$\begin{aligned} \|y_n - \hat{y}_n\|^2 &= (y_n - \hat{y}_n)^T (y_n - \hat{y}_n) \\ &= y_n^T y_n - 2y_n^T \hat{y}_n + \hat{y}_n^T \hat{y}_n \\ &= y_n^T y_n - 2\sum_{i=1}^{K} y_n^i P(C_i|x_n) + \sum_{j=1}^{K} (P(C_j|x_n))^2 \\ &= y_n^T y_n + \sum_{j=1}^{K} [(P(C_j|x_n))^2 - 2y_n^j P(C_j|x_n)] . \end{aligned} \tag{4}$$

Thus, we have

$$\nabla_{\Theta}\|y_n - \hat{y}_n\|^2 = 2\sum_{j=1}^{K}[(P(C_j|x_n) - y_n^j)\nabla_{\Theta}P(C_j|x_n)] \tag{5}$$

where Θ is μ_k^i or $(\Sigma_k^i)^{-1}$ (for computational convenience, we use $(\Sigma_k^i)^{-1}$ instead of Σ_k^i from now on) in this paper.

2.2 Derive $\nabla_{\mu_k^i} P(C_j|x_n)$

Define $\Phi_1 = \frac{P(C_k)a_k^i}{p(x_n)} G(x_n|\mu_k^i, \Sigma_k^i)(\Sigma_k^i)^{-1}(x_n - \mu_k^i)$, then

$$\nabla_{\mu_k^i} P(C_j|x_n) = \begin{cases} [1 - P(C_k|x_n)]\Phi_1, & for\ j = k \\ -P(C_j|x_n)\Phi_1, & for\ j \neq k \end{cases} \tag{6}$$

(see Appendix A for details).

2.3 Derive $\nabla_{(\Sigma_k^i)^{-1}} P(C_j|x_n)$

Because $\nabla_{\Sigma_k^i} P(C_j|x_n)$ is difficult to get directly, we try to derive $\nabla_{(\Sigma_k^i)^{-1}} P(C_j|x_n)$ alternatively.

$$\nabla_{(\Sigma_k^i)^{-1}} P(C_j|x_n) = \begin{cases} \frac{P(C_k)a_k^i}{p(x_n)}[1 - P(C_k|x_n)]\Phi_2, & for\ j = k \\ -\frac{P(C_k)a_k^i P(C_j|x_n)}{p(x_n)}\Phi_2, & for\ j \neq k \end{cases} \tag{7}$$

where

$$\Phi_2 = G(x_n|\mu_k^i, \Sigma_k^i)\{\Sigma_k^i - \frac{1}{2}diag(\Sigma_k^i) - A + \frac{1}{2}diag(A)\}$$

with $A = (x_n - \mu_k^i)(x_n - \mu_k^i)^\top$ (see Appendix B for details).

2.4 Decorrelated LMS Algorithm for Bayessian Classifier

With the derived stochastic gradient formulation in (5), one might seek to update parameter Θ using the gradient directly (namely LMS algorithm), i.e. using

$$\Theta_n = \Theta_{n-1} - \mu_n \nabla_{\Theta_{n-1}} \|y_n - \hat{y}_n\|^2 \tag{8}$$

to carry out on-line learning adaptively, where μ_n is the learning rate [15]. However, this would take a risk of low convergence rate and poor tracking performance, since stochastic gradient $\nabla_{\Theta_{n-1}}\|y_n - \hat{y}_n\|^2$ is only the instantaneous approximation of the true gradient which should be derived from $\nabla_{\Theta_{n-1}} E\{\|y_n - \hat{y}_n\|^2\}$. If two consecutive instantaneous gradients correlate with each other, then the mean square error (MSE) might be accumulated and couldn't be corrected in time. Therefore, to get rid of these shortcomings, here we adopt the decorrelated gradient instead of the instantaneous gradient [15][16]. Using decorrelated

Table 1. The flow chart of the decorrelated LMS (DLMS) algorithm for learning on-line classification

The variable Θ in the following procedure denotes μ_k^i or $(\Sigma_k^i)^{-1}$ with $\{k = 1, \ldots, K; i = 1, \ldots, N_k\}$.
Step 1:
Initialize Θ with Θ_0.
Step 2:
For $n = 1, 2, \ldots$, calculate the decorrelated gradient $\hat{\nabla}_{\Theta_{n-1}} \|y_n - \hat{y}_n\|^2$ from (5) and (9), and update Θ with $\Theta_n = \Theta_{n-1} - \mu_n \hat{\nabla}_{\Theta_{n-1}} \|y_n - \hat{y}_n\|^2$.

gradient can effectively avoid the case of error accumulation which might arise in instantaneous gradient descent algorithms, and hence, can accelerate the convergence of the adaptive-gradient methods.

The decorrelated gradient of Θ_n can be defined as

$$\hat{\nabla}_{\Theta_n} \|y_n - \hat{y}_n\|^2 = \nabla_{\Theta_n} \|y_n - \hat{y}_n\|^2 - a_n \nabla_{\Theta_{n-1}} \|y_n - \hat{y}_n\|^2 \tag{9}$$

where a_n is the decorrelation coefficient between $\nabla_{\Theta_n} \|y_n - \hat{y}_n\|^2$ and $\nabla_{\Theta_{n-1}} \|y_n - \hat{y}_n\|^2$. For two vectors v_n and v_{n-1}, the decorrelation coefficient a_n can be defined as

$$a_n = \frac{(v_n - \bar{v}_n)^\top (v_{n-1} - \bar{v}_{n-1})}{(v_{n-1} - \bar{v}_{n-1})^\top (v_{n-1} - \bar{v}_{n-1})} \tag{10}$$

where $\bar{v}_n$ represents the mean value of v_n [15]. For two matrices, the concept of decorrelation coefficient can be similarly extended. Table 1 describes the paradigm of our proposed decorrelated LMS (DLMS) algorithm for learning on-line classification.

3 Experiments

3.1 Materials and Protocols

Here we describe the data set analyzed in this paper. The data set contains EEG recordings from 3 normal subjects (denoted by A, B, C respectively) during non-feedback mental imagery tasks. The subjects sat in a normal chair, relaxed arms resting on their legs. The three tasks are: imagination of repetitive self-paced left hand movements (class C_1), imagination of repetitive self-paced right hand movements (class C_2) and generation of different words beginning with the same random letter (class C_3).

For a given subject, there are 3 recording sessions acquired on the same day, each lasting about 4 minutes with breaks of 5-10 minutes in between. The subject performed a given task for about 15 seconds and then switched randomly to the next task at the operator's request. The raw EEG potentials were first spatially filtered by means of a surface Laplacian [17][18]. The superiority of

surface Laplacian transformation over raw potentials for the operation of BCI has already been demonstrated [19]. Then, every 62.5 ms, the power spectral density in the band 8-30Hz was estimated over the last second of data with a frequency resolution of 2 Hz for 8 centro-parietal channels (EEG signals recorded over this region reflects the activities of brain's sensorimotor cortices). The power spectra in the frequency band 8-30 Hz were then normalized according to the total energy in that band. As a result, an EEG sample is a 96-dimensional vector (8 channels times 12 frequency components). The total number of samples for subjects A, B, and C during three sessions are respectively 3488/3472/3568, 3472/3456/3472, and 3424/3424/3440. For a more detailed description of the data and the brain computer interface protocol, please refer to [7]. In this article, we concentrate on utilizing the 96 dimensional pre-computed features to address the problem of on-line classification.

3.2 Experimental Results

EEG signal classification is conducted for each subject. First of all, to reduce the parameters to be estimated and avoid the over-fitting problem, principal component analysis (PCA) is adopted to reduce the feature dimensions by reserving 90% energy. The threshold 90% is a good tradeoff between dimension reduction and energy preservation for our problem. To initialize the parameters μ_k^i and Σ_k^i of the DLMS algorithm, we first apply the k-Means clustering algorithm with multiple runs [10], and the result with the least cost value is selected for initialization utility. On the selection of parameters $P(C_k)$, N_k and a_k^i in the Bayesian classifier of Gaussian mixture models, we take the same configuration as [7], for in his research, Millán had shown its effectiveness through cross-validations. Thus, $P(C_k) = \frac{1}{3}$, $N_k = 4$ and $a_k^i = \frac{1}{4}$ $(k = 1, 2, 3; i = 1, 2, 3, 4)$.

In this article, the data of session 1 from each mental task of every subject is employed to implement parameter initialization. For class C_k, we first use k-Means clustering algorithm to initialize μ_k^i which comes from one of the N_k cluster centers. Then Σ_k^i can be obtained using the data belonging to the same cluster C_k^i. Subsequently, we update the parameters adaptively on the first one minute data of the next session (the samples are processed sequentially and only once, to completely stimulate the on-line situation). With the final updated parameters, we test the performance of the classifier on the data of the last three minutes from the next session. The learning rate of μ_k^i and $(\Sigma_k^i)^{-1}$ are taken as 1e-6 and 1e-4 respectively, which are found to provide good classification results among a small number of parameter search for the basic LMS algorithm. The same procedure is performed on session 2 and session 3, i.e., we initialize the parameters μ_k^i and Σ_k^i through k-Means clustering on session 2, then update them using the first one minute data of session 3 and test the final classifier on the last three minute data of session 3.

To evaluate the performance of our decorrelated LMS (DLMS) algorithm for learning on-line classification, under the same conditions we also carry out on-line classification using the basic LMS algorithm, which adopts instantaneous gradient instead of decorrelated gradient to update parameters. The final

Table 2. Classification accuracies of on-line learning by LMS algorithm and decorrelated LMS (DLMS) algorithm

Subjects	Sessions	LMS	DLMS
A	2	67.79%	67.87%
	3	70.71%	70.59%
B	2	47.40%	45.63%
	3	51.83%	52.31%
C	2	49.19%	48.78%
	3	41.45%	42.82%

classification accuracy rates using these two classifiers with parameters updated by the whole one minute data are given in Table 2.

Through statistical Z-test, no significant difference is found between the final results of these two algorithms (p-value=0.8845). This only indicates that the

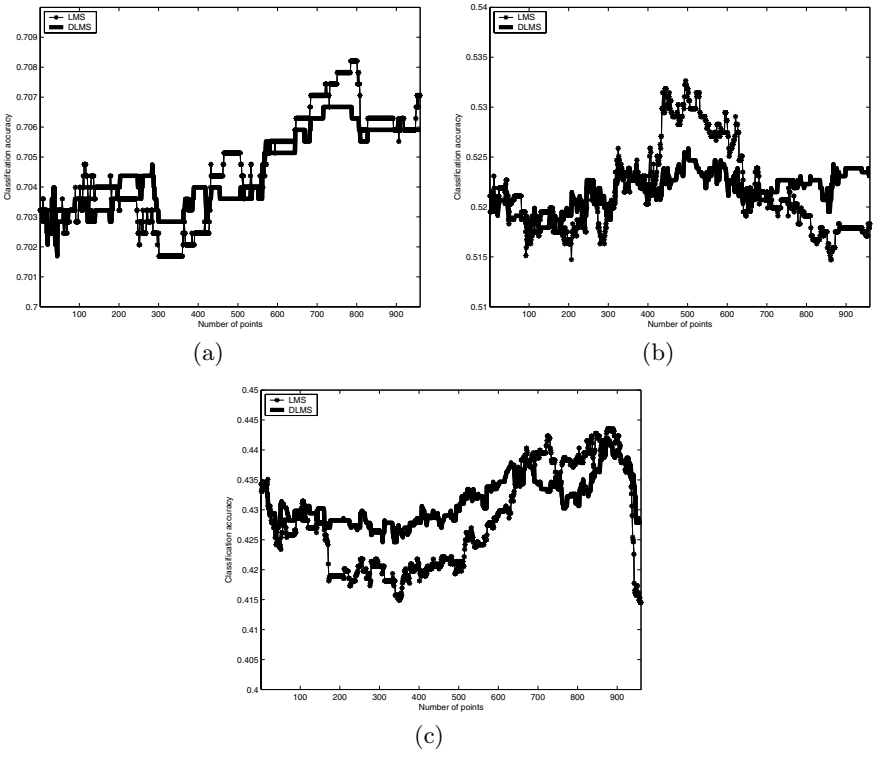

Fig. 1. (a): The time course of classification accuracies on session 3, subject A. (b): The time course of classification accuracies on session 3, subject B. (c): The time course of classification accuracies on session 3, subject C.

Table 3. The standard deviations (STDs) (normalized to the range [1, 10]) of the time courses of on-line classification by LMS algorithm and DLMS algorithm

Subjects	Sessions	LMS STD	DLMS STD	STD Reduced
A	2	1.46	1.02	30.1%
	3	1.79	1.30	27.4%
B	2	9.95	2.22	77.7%
	3	4.32	1.87	56.7%
C	2	4.26	4.42	−3.8%
	3	8.39	3.89	53.6%
Average				40.28%

performance of LMS algorithm is statistically similar to DLMS algorithm after a long time of update. As we have stated before, one important requirement for on-line BCI applications is to improve the classification performance using as minimal training data as possible. Besides, for non-feedback BCIs, as there is no sign helping subjects to rectify their latent strategies generating EEG signals, effective algorithms should be of good stability. Below we give the time courses of the convergence of these two algorithms during the on-line update stage for classifying the last three minutes of session 3 of three subjects in Fig. 1. That is, after every update, we obtain the classification accuracy on the last 3 minutes of session 3. From Fig. 1, the robustness and the rapid convergence of DLMS algorithm are manifested. Although the results of LMS algorithm and DLMS algorithm have the same tendencies, the magnitude variance of classification accuracy obtained by DLMS algorithm is rather smaller than that of LMS algorithm. Thus the rapid convergence and robustness of DLMS algorithm are indicated. For other test sessions, similar results are observed. In addition, to give a quantitative description, the standard deviations of the classification results for LMS algorithm and DLMS algorithm are respectively given in Table 3, from which we can see that by using DLMS algorithm for gradient descent the standard deviation has been reduced to a large extent.

4 Related Work

With regard to the idea of stochastic gradient descent, Millán *et al.,* have mentioned it in their publications [7][11][13]. However, they usually make a very rigorous assumption about the formulation of covariance matrix Σ_k^i, such as the assumption of diagonal and common to all the prototypes of a certain class, and make a simple approximation about the gradient of μ_k^i. Fig. 2 shows the distribution of two features from the original 96 ones. Clearly, using the combination of diagonal covariances could not represent the external oblique distribution logically.

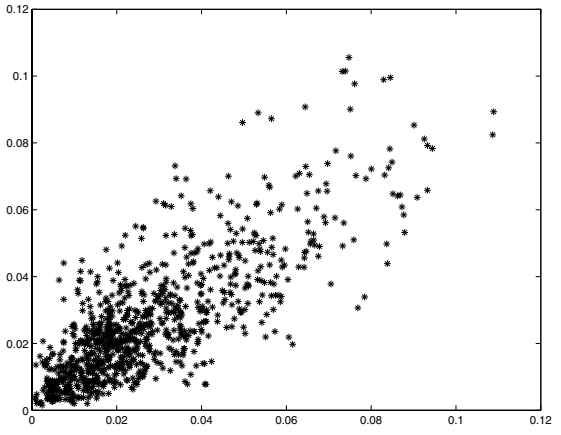

Fig. 2. The distribution of two features from the original 96 ones of session 1, subject A

While in this article, PCA is adopted for dimension reduction and the covariance matrices are described with a general form. This would be more reasonable and more powerful in depicting different data distributions. Besides, we independently derive the general representation of gradient descent algorithm for μ_k^i and $(\Sigma_k^i)^{-1}$ in a Bayesian classifier context, which didn't appear before in the literature as far as we know. In addition, a new algorithm namely decorrelated LMS algorithm is proposed for the on-line learning of μ_k^i and $(\Sigma_k^i)^{-1}$, and obtains better performance than the basic LMS algorithm (stochastic gradient descent algorithm). These make our current work much different from Millán's.

5 Conclusions and Future Work

The research of brain-computer interface technology is an interdisciplinary project, which gestates many challenges in a variety of aspects. In this paper, we address the problem of on-line classification of EEG signals with applications to brain-computer interfaces. The time-varying characteristic of EEG recordings between experimental sessions makes it a difficult issue to categorize different EEG signals, and necessitates learning the on-line classification. Based on a Bayesian classifier of Gaussian mixture models, we derive the general formulations of the instantaneous gradient and the decorrelated gradient. Besides, a decorrelated LMS algorithm (DLMS) is developed to accelerate the convergence of the traditional LMS algorithm (stochastic gradient descent method). Experiments and comparisons shows the effectiveness and robustness of our approach.

For practical utilities, one can design a easy-going protocol to implement on-line learning. Each time users make use of BCI equipments after a long break, there would be a on-line learning stage of one minute or so during which a display device generates a series of random signs indicating upcoming tasks. Following these cues, users carry out specific mental activities. Simultaneously,

the classifier would be updated on-line. In the future, study on the realization of automatic on-line training and on the active selection of training instances would be an interesting issue.

Acknowledgements

Shiliang Sun and Changshui Zhang would like to thank IDIAP Research Institute (Switzerland) for providing the analyzed data. Besides, we would also like to thank the National Natural Science Foundation of China for supporting this work under Project 60475001.

References

1. Nicolelis, M.A.L.: Actions from Thoughts. Nature, Vol. 409 (2001) 403-407
2. Wolpaw, J.R., Birbaumer, N., McFarland, D.J., Pfurtscheller, G., Vaughan, T.M.: Brain-Computer Interfaces for Communication and Control. Clinical Neurophysiology, Vol. 113 (2002) 767-791
3. Ebrahimi, T., Vesin, J.M., Garcia, G.: Brain-Computer Interfaces in Multimedia Communication. IEEE Signal Processing Magazine, Vol. 20 (2003) 14-24
4. Wickelgren, I.: Tapping the Mind. Science, 299 (2003) 496-499
5. Wolpaw, J.R., Birbaumer, N., Heetderks, W.J., McFarland, D.J., Peckham, P.H., Schalk, G., Donchin E., Quatrano, L.A., Robinson, C.J., Vaughan, T.M.: Brain-Computer Interface Technology: A Review of the First International Meeting. IEEE Transactions on Rehabilitation Engineering, Vol. 8 (2000) 164-173
6. Vaughan, T.M.: Guest Editorial Brain-Computer Interface Technology: A Review of the Second International Meeting. IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 11 (2003) 94-109
7. Millán, J.R.: On the Need for On-Line Learning in Brain-Computer Interfaces. Proceedings of 2004 International Joint Conference on Neural Networks. Budapest, Hungary (2004)
8. Mitchell, T.M.: Machine Learning. McGraw-Hill, New York (1997)
9. Saad, D.: On-Line Learning in Neural Networks. Cambridge University Press (1998)
10. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. 2th edn. John Wiley & Sons, New York (2000)
11. Millán, J.R., Renkens, F., Mouriño, J., Gerstner, W.: Non-Invasive Brain-Actuated Control of a Mobile Robot. Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, (2003) 1121-1126
12. Millán, J.R., Renkens, F., Mouriño, J., Gerstner, W.: Brain-Actuated Interaction. Artificial Intelligence, Vol. 159 (2004) 241-259
13. Millán, J.R., Renkens, F., Mouriño, J., Gerstner, W.: Noninvasive Brain-Actuated Control of a Mobile Robot by Human EEG. IEEE Transactions on Biomedical Engineering, Vol. 51 (2004) 1026-1033
14. Mclachlan, G., Peel, D.: Finite Mixture Models. Wiley, New York (2000)
15. Glentis, G.O., Berberidis, K., Theodoridis, S.: Efficient Least Square Adaptive Algorithms for FIR Transversal Filtering. IEEE Signal Processing Magzine, Vol. 16 (1999), 13-41

16. Doherty, J., Porayath, R.: A Robust Echo Canceler for Acoustic Environments. IEEE Transactions on Circuits and Systems, II, Vol, 44 (1997) 389-398
17. Perrin, R., Pernier, J., Bertrand, O., Echallier, J.: Spherical Spline for Potential and Current Density Mapping. Electroencephalography and Clinical Neurophysiology, Vol. 72 (1989), 184-187
18. Perrin, R., Pernier, J., Bertrand, O., Echallier, J.: Corrigendum EEG 02274. Electroencephalography and Clinical Neurophysiology, Vol. 76 (1990), 565
19. McFarland, D.J., McCane, L.M., David, S.V., Wolpaw, J.R.: Spatial Filter Selection for EEG-Based Communication. Electroencephalography and Clinical Neurophysiology, Vol. 103 (1997) 386-394

Appendix A: Derive $\nabla_{\mu_k^i} P(C_j|x_n)$

$$\begin{aligned}\nabla_{\mu_k^i} P(C_j|x_n) &= \nabla_{\mu_k^i} \frac{P(C_j)p(x_n|C_j)}{p(x_n)} \\ &= \nabla_{\mu_k^i} \frac{P(C_j)\sum_{l=1}^{N_j} a_j^l G(x_n|\mu_j^l, \Sigma_j^l)}{p(x_n)} \end{aligned} \tag{11}$$

A.1 When $j = k$

$$\begin{aligned}&\nabla_{\mu_k^i} \frac{P(C_j)\sum_{l=1}^{N_j} a_j^l G(x_n|\mu_j^l, \Sigma_j^l)}{p(x_n)} \\ &= \frac{P(C_k)a_k^i}{p(x_n)}[1 - P(C_k|x_n)]\nabla_{\mu_k^i} G(x_n|\mu_k^i, \Sigma_k^i)\end{aligned} \tag{12}$$

where

$$\nabla_{\mu_k^i} G(x_n|\mu_k^i, \Sigma_k^i) = G(x_n|\mu_k^i, \Sigma_k^i)(\Sigma_k^i)^{-1}(x_n - \mu_k^i)\ . \tag{13}$$

A.2 When $j \neq k$

$$\begin{aligned}&\nabla_{\mu_k^i} \frac{P(C_j)\sum_{l=1}^{N_j} a_j^l G(x_n|\mu_j^l, \Sigma_j^l)}{p(x_n)} \\ &= -\frac{P(C_k)a_k^i P(C_j|x_n)}{p(x_n)} \nabla_{\mu_k^i} G(x_n|\mu_k^i, \Sigma_k^i)\ .\end{aligned} \tag{14}$$

Appendix B: Derive $\nabla_{(\Sigma_k^i)^{-1}} P(C_j|x_n)$

$$\begin{aligned}\nabla_{(\Sigma_k^i)^{-1}} P(C_j|x_n) &= \nabla_{(\Sigma_k^i)^{-1}} \frac{P(C_j)p(x_n|C_j)}{p(x_n)} \\ &= \nabla_{(\Sigma_k^i)^{-1}} \frac{P(C_j)\sum_{l=1}^{N_j} a_j^l G(x_n|\mu_j^l, \Sigma_j^l)}{p(x_n)} \end{aligned} \tag{15}$$

B.1 When $j = k$

$$\begin{aligned}&\nabla_{(\Sigma_k^i)^{-1}} \frac{P(C_j)\sum_{l=1}^{N_j} a_j^l G(x_n|\mu_j^l, \Sigma_j^l)}{p(x_n)} \\ &= \frac{P(C_k)a_k^i}{p(x_n)}[1 - P(C_k|x_n)]\nabla_{(\Sigma_k^i)^{-1}} G(x_n|\mu_k^i, \Sigma_k^i)\ . \end{aligned} \tag{16}$$

Considering the normal distribution $G(x_n|\mu_k^i, \Sigma_k^i) = \frac{1}{(2\pi)^{d/2}|\Sigma_k^i|^{1/2}} exp\{-\frac{1}{2}(x_n - \mu_k^i)^\top (\Sigma_k^i)^{-1}(x_n - \mu_k^i)\} = \frac{1}{(2\pi)^{d/2}|\Sigma_k^i|^{1/2}} exp\{-\frac{1}{2} tr[(\Sigma_k^i)^{-1}(x_n - \mu_k^i)(x_n - \mu_k^i)^\top]\}$, if we denote $A = (x_n - \mu_k^i)(x_n - \mu_k^i)^T$, then

$$\begin{aligned}&\nabla_{(\Sigma_k^i)^{-1}} G(x_n|\mu_k^i, \Sigma_k^i) \\ &= \frac{1}{(2\pi)^{d/2}} exp\{-\frac{1}{2} tr[(\Sigma_k^i)^{-1}A]\}\frac{1}{2}|(\Sigma_k^i)^{-1}|^{-\frac{1}{2}}|(\Sigma_k^i)^{-1}|[2\Sigma_k^i - diag(\Sigma_k^i)] + \\ &\quad G(x_n|\mu_k^i, \Sigma_k^i)\{-\frac{1}{2}[2A - diag(A)]\} \\ &= G(x_n|\mu_k^i, \Sigma_k^i)\{\Sigma_k^i - \frac{1}{2}diag(\Sigma_k^i) - A + \frac{1}{2}diag(A)\}. \end{aligned} \tag{17}$$

B.2 When $j \neq k$

$$\begin{aligned}&\nabla_{(\Sigma_k^i)^{-1}} \frac{P(C_j)\sum_{l=1}^{N_j} a_j^l G(x_n|\mu_j^l, \Sigma_j^l)}{p(x_n)} \\ &= -\frac{P(C_k)a_k^i P(C_j|x_n)}{p(x_n)} \nabla_{(\Sigma_k^i)^{-1}} G(x_n|\mu_k^i, \Sigma_k^i). \end{aligned} \tag{18}$$