ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

Evidence build-up facilitates on-line adaptivity in dynamic environments: example of the BCI P300-speller

Emmanuel Daucé[1,2] and Eoin Thomas[3]

1- INSERM UMR 1106 - Institut de Neuroscience des Systèmes
Faculté de Médecine de la Timone - Aix-Marseille Université - France

2- Centrale Marseille, France

3- INRIA Lille Nord Europe, France

Abstract. We consider a P300 BCI application where the subjects can write figures and letters in an unsupervised fashion. We (i) show that a generic speller can attain the state-of-the-art accuracy without any training phase or calibration and (ii) present an adaptive setup that consistently increases the bit rate for most of the subjects.

1 Introduction

In reactive Brain-Computer Interfaces (BCI), a series of stimuli is displayed while the subject's EEG responses (Event Related Potentials - ERP) are recorded. For instance, when the stimulus displayed on the screen depart from the other stimuli of the series, a particular ERP -called the "P300"- is produced, that correlates with to the subject's "perception of change". This subtle change on the EEG can be detected and used in a speller setup [1], where sequences of symbols are displayed on the screen and the subject is asked to keep concentrated on the particular symbol he wants to spell out. In the most common setup , the subject faces a screen with a 6×6 grid of letters and numbers while rows and columns of the grid are flashed at random and EEG samples are extracted. Then, the classifier has to identify in the set the index of the row and column where the "P300" took place, so that the selected character is at the intersection of this row and this column. Then another series of flashes starts over until the expected sequence of letters is formed.

The evidence accumulation scheme (or sequential analysis [2]), has received a recent interest in the field. There, a typical bottleneck is the time needed for issuing a reliable command, where the speed/accuracy trade-off is measured in terms of a bit rate [3] which represents the number of bits that can be transmitted per minute. When the data is clean, only few examples may be necessary to reach this target accuracy. On the contrary, when the data is noisy, more data should be needed to reach the same accuracy. We consider in section 2 such a recursive evidence accumulation allowing to optimize the speed-accuracy trade-off in a very explicit manner. Then we consider in section 3 the non stationary case when

This research was partly supported by french ANR "Défis" project CO-ADAPT (ANR-09-EMER-002). EEG data kindly provided by Jérémie Mattout and Emmanuel Maby from INSERM U1028.

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

unforeseen changes take place in the environments. The classifier is expected to *adapt* (update its parameters) in order to prevent the decrease of its accuracy. When the environment does not change too fast, a simple stochastic gradient should be implemented, so that the changes should be smoothly followed, using a semi-supervised [4] gradient descent.

2 Evidence accumulation in the "oddball" case

The EEG dataset we use comes from a P300 experiment reported in [5]. The data consists of 20 files, one file per subject measuring the brain activity during a P300-speller experiment where each subject had to spell out mentally 110 letters.

Let $\mathcal{A}$ be a set of symbols. In order to guess the expected symbol $y^* \in \mathcal{A}$, a sequence of random subsets of $\mathcal{A}$ is flashed on the screen. We suppose in the following that each subset contains on average L/K symbols, where L is the cardinal of $\mathcal{A}$, so that every symbol appears on average once in a series of K flashes. A particular ERP is expected to take place when the target symbol is flashed. We note $S_1, ..., S_T$ this sequence of subsets. For every $t \in \{1, ..., T\}$, an observation vector $\boldsymbol{x}_t$ is extracted from the EEG signal.

A probabilistic classifier is a classifier whose response is a probability over the number of classes defined in the classification problem. In our case, we consider a classifier which has been trained on the basis of single trial examples so that the output $f(\boldsymbol{x}_t)$ approximates $P(y^* \in \mathcal{S}_t|\boldsymbol{x}_t)$.

In a Bayesian framework, one can consider the classifier implements the formula:

$$f(\boldsymbol{x}_t) \simeq P(y^* \in \mathcal{S}_t|\boldsymbol{x}_t) = \frac{\rho P(\boldsymbol{x}_t|y^* \in \mathcal{S}_t)}{\rho P(\boldsymbol{x}_t|y^* \in \mathcal{S}_t) + (1-\rho)P(\boldsymbol{x}_t|y^* \notin \mathcal{S}_t)} \tag{1}$$

where the target vectors are distributed according to $P(.|y^* \in \mathcal{S})$ while the non-target vectors are distributed according to $P(.|y^* \notin \mathcal{S})$, and $\rho = P(y^* \in \mathcal{S})$ is the density of symbols in one flash.

Now the point is to calculate the probabilities $\pi(\alpha|\mathbf{x}_{1:t})$'s for each symbol and at each flash t (with $\sum_{\alpha\in\mathcal{A}} \pi(\alpha|\mathbf{x}_{1:t}) = 1$). At each flash, one can decide to continue flashing or to stop if there is enough evidence for a given symbol. The stopping criterion is a threshold based on a confidence level. For instance, with a 90% confidence level, the stopping criterion will be $\exists\alpha : \pi(\alpha|\mathbf{x}_{1:t}) > 0.9$.

After observing $\boldsymbol{x}_t$, the instantaneous probability for each symbol is

$$\begin{aligned}
\pi(\alpha|\boldsymbol{x}_t) \simeq & \quad P(y^* = \alpha|\boldsymbol{x}_t) \\
= & \quad \mathbf{1}_{\alpha\in\mathcal{S}_t} P(y^* = \alpha|y^* \in \mathcal{S}_t)P(y^* \in \mathcal{S}_t|\boldsymbol{x}_t) \\
& + \mathbf{1}_{\alpha\notin\mathcal{S}_t} P(y^* = \alpha|y^* \notin \mathcal{S}_t)P(y^* \notin \mathcal{S}_t|\boldsymbol{x}_t)
\end{aligned}$$

where

$$P(y^* = \alpha|y^* \in \mathcal{S}) = \frac{1}{|\mathcal{S}|}$$

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

so that

$$\pi(\alpha|\boldsymbol{x}_t) = \mathbf{1}_{\alpha \in \mathcal{S}_t} \frac{f(\boldsymbol{x}_t)}{|\mathcal{S}_t|} + \mathbf{1}_{\alpha \notin \mathcal{S}_t} \frac{1 - f(\boldsymbol{x}_t)}{|\mathcal{A} \smallsetminus \mathcal{S}_t|}$$

and

$$\pi(\alpha|\mathbf{x}_{1:t}) = \frac{\pi(\alpha|\boldsymbol{x}_t)\pi(\alpha|\mathbf{x}_{1:t-1})}{\sum_{\beta \in \mathcal{A}} \pi(\beta|\boldsymbol{x}_t)\pi(\beta|\mathbf{x}_{1:t-1})}$$

where a log-posterior rescaling scheme [6] is used in order to avoid numerical instabilities.

Linear-Gaussian implementation A critical aspect of our cumulative-adaptive setup is the correctness of the posterior calculation. Consistently with the subject-independent approach proposed in [7], we cross-validate our classifier by learning it on 19 subjects and testing it on the remaining subject (and repeat the same operation for every subject). In our case, the number of samples (264,000) largely overtakes the vectors dimension (1920), which allows to adopt a close-form solution to the optimization problem using a light regularization $\lambda = 1e^{-4}$. The explicit posterior estimates is then :

$$f(x) = \frac{1}{1 + \exp(-\boldsymbol{x}\boldsymbol{w}^T - b)} \tag{2}$$

with $\boldsymbol{w} = (\boldsymbol{\mu}_{\mathbf{1}} - \boldsymbol{\mu}_{\mathbf{0}})(\boldsymbol{\Sigma} + \lambda \boldsymbol{I})^{-1}$ and $b = -0.5\boldsymbol{\mu}_1(\boldsymbol{\Sigma} + \lambda \boldsymbol{I})^{-1}\boldsymbol{\mu}_1^T + 0.5\boldsymbol{\mu}_0(\boldsymbol{\Sigma} + \lambda \boldsymbol{I})^{-1}\boldsymbol{\mu}_0^T - \ln(K-1)$, where $\boldsymbol{\mu}_{\mathbf{1}}$ is the target class average, $\boldsymbol{\mu}_{\mathbf{0}}$ is the non-target class average and $\boldsymbol{\Sigma}$ is the shared covariance matrix [8].

The passing from the binary classification to the 36-symbols classification is done at the cost of several flash repetitions, where every flash brings a new information that takes part in the accumulation. With our dataset, we measure after each series of 12 flashes wether the evidence on one symbol reaches the threshold corresponding to a character posterior probability of 90%, in which case the response is given and the next letter is considered. When the threshold is not reached after 12×10 flashes, the MAP choice is applied nonetheless and the next letter considered. We give in table 1 the different spelling accuracies, the mean number of flashes and the bit rate[1] obtained for every subject of the dataset using the regularized multivariate Gaussian classifier $(\boldsymbol{w}, b)$.

The spelling result appear generally good, with an average spelling accuracy (85.2 %) that closely compares to most of the existing subject-specific spellers, which is surprisingly good, if we remember that our generic classifier has been learned on a different set of subjects than the one over which it is tested. A strong discrepancy across the different subjects is however to be noticed, with individual accuracies ranging from 15% to 100%. The bit rate also appears extremely variable, ranging from 0.6 bits min^{-1} to 36.5 bits min^{-1}. This subject

[1] The number of bits per decision being $I = \log_2 L + r \log_2 r + (1-r)\log_2 \frac{1-r}{L-1}$, where L is the number of symbols and r is the spelling accuracy, the bit rate is $I \times D$, where D is the (average) number of decisons per minute.

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

subject	1	2	3	4	5	6	7	8	9	10	
spel. acc.	15.4	94.5	99.1	100	97.3	93.6	96.4	83.6	99.1	47.3	
avg. flash #	105	69	47	58	68	61	55	70	55	92	
bit rate	0.6	20	32.1	26.9	21.5	22.1	26	15.8	27.4	4.7	
subject	11	12	13	14	15	16	17	18	19	20	**all**
spel. acc.	71.8	65.4	100	98.2	100	97.3	88.2	86.4	82.7	87.3	**85.2**
avg. flash #	86	99	42	47	55	59	75	74	79	85	**69**
bit rate	9.9	7.4	36.5	31.5	27.7	24.6	16.2	15.9	13.7	13.9	**19.7**

Table 1: Average spelling accuracies, average number of flashes and average bit rate obtained for every subject under the evidence accumulation scheme.

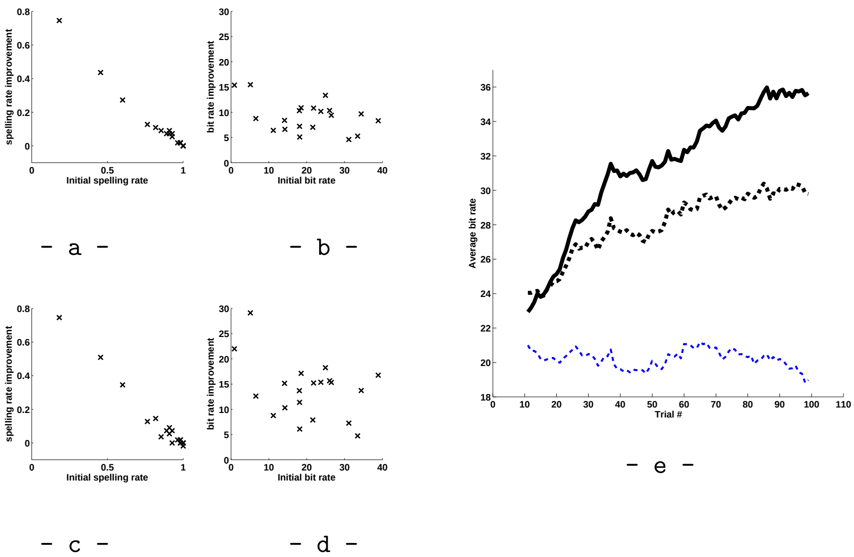

Fig. 1: Accuracy and bit rate improvement with the online update. **- a -** Spelling accuracy improvement with the standard gradient descent, measured for each subject on the last 55 trials. The initial accuracy is on the x-axis and the difference on the y-axis. **- b -** Bit rate improvement with the standard gradient descent. **- c -** Spelling improvement for the reiterated gradient descent. **- d -** Bit rate improvement for the reiterated gradient descent. **- e -** Bit rate improvement trend, averaged on all the subjects, with the reiterated gradient descent (bold line), the standard gradient descent (dashed bold line) and without online update (thin dashed line).

discrepancy seems at first seek to be higher than in the subject-specific case. If common features are clearly developped across the different subjects, subject-specific discriminating features need to be considered in order to optimize the spelling accuracy. Those preliminary results show that acceptable "ready-to-

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7. Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

use" P300 spellers are feasible at the cost of learning the generic classifier on a large and representative enough database.

3 On-line adaptation

The second learning phase we defined consists in updating the classifier in a way that should better fit the subject's own characteristics. The principles at stake are again very straightforward, relying on a standard stochastic gradient descent. Having the subject-independent model as initial guess, our contribution is to exploit the labels obtained from the early-stopping accumulation process to guide the gradient descent. A virtuous circle between model and label improvement is thus expected, taking for granted that few labelling errors should not perturbate the general gradient direction.

The second hyperparameter used here is the learning rate η, and, in accordance with (2), the update is made after every response, when a new set of labelled examples $((\boldsymbol{x}_1, \hat{y}), ..., (\boldsymbol{x}_{\hat{t}}, \hat{y}))$ is available, where $\hat{t}$ is the number of the flash at which the accumulation was stopped. Starting from $\boldsymbol{w}_0 = \boldsymbol{w}$, $b_0 = b$, and for t in $1..\hat{t}$:

$$\boldsymbol{w}_t = (1 - \eta\lambda)\boldsymbol{w}_{t-1} + \eta(\mathbf{1}_{\hat{y} \in \mathcal{S}_t} - f(\boldsymbol{x}_t))\boldsymbol{x}_t \tag{3}$$

$$b_t = (1 - \eta\lambda)b_{t-1} + \eta(\mathbf{1}_{\hat{y} \in \mathcal{S}_t} - f(\boldsymbol{x}_t)) \tag{4}$$

Then update the parameters using $\boldsymbol{w} \leftarrow \boldsymbol{w}_{\hat{t}}$ and $b \leftarrow b_{\hat{t}}$.

In a variant of the scheme, we keep in memory the data of the 50 most recent trials, and replay the accumulation procedure on past examples in order to recalculate both the posteriors and the labels, so that the past labels may be reconsidered at the light of the most recent updates. The same data being passed several times, the gradient descent should be accelerated with the counterpart of a higher computational cost. We compare the effectiveness of the two methods on the 20 subjects, using the same linear-Gaussian classifiers as in the evaluation, and taking $\eta = 0.1$ in the standard gradient descent and $\eta = 0.01$ in the reiterated gradient descent. The results are displayed on figure 1.

Here again, the improvement obtained appears quite good and robust, while not equally beneficial between subjects. Considering figure 1 - a -, representing the spelling improvement in function of the initial accuracy, a linear effect seems to draw out, with a greater benefit for the subjects having initially the weaker spelling accuracies, which is rather natural since a higher spelling accuracy has little room to improve. The improvement in the bit rate is more varied across subjects (1 - b -), ranging from strong improvement (about 15 bits/min) to lesser 5 bits/min improvement, independently of the initial bit rate. The bit rate improvement, interpreted as a shortening of the decision process, is not obtained at the detriment of the spelling accuracy. It seems though that additional improvement may be possible, which is confirmed by the results obtained with the reiterated gradient descent (fig 1 - c - and - d -), with a more consistant bit rate increase and a similar spelling accuracy increase. The general trend obtained in figure 1 - e - confirms the continuing tendency toward bit rate improvement,

ESANN 2014 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 23-25 April 2014, i6doc.com publ., ISBN 978-287419095-7.
Available from http://www.i6doc.com/fr/livre/?GCOI=28001100432440.

with a stronger effect when the reiterated gradient is used. Most of the gain is obtained on the speed of the response, with a spelling accuracy converging around 90 %, which is consistent with the considered accuracy threshold.

4 Conclusions

We have presented a general evidence accumulation scheme to be used in the context of online decision making with multiple observations in noisy and changing environments. With the appropriate probabilistic classifiers, a fine tuning of the accuracy can be done, allowing to optimize the speed-accuracy trade-off in a very explicit manner. In a context of unsupervised online classification, the accumulation method allows to put a label on the new data with a confidence that matches the target accuracy. We illustrate this principle on a practical BCI application where the subjects have to write figures and letters in an unsupervised fashion. We show, accordingly with [7], that a generic speller can attain the state-of-the-art accuracy without any training phase or calibration. We also show that our online adaptive setup is beneficial for every subject considered in this study, with a stronger improvement for the subjects having the lowest initial spelling accuracy. The results have been obtained in simulation, but the quite simple principles at stake should make it possible to test it real experiments and in more realistic life-long usage.

References

[1] L.A. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510 – 523, 1988.

[2] A Wald. *Sequential analysis*. Dover Phoenix Editions, Mineola, NY, 1947.

[3] Bernardo Dal Seno, Matteo Matteucci, and Luca Mainardi. Online detection of p300 and error potentials in a bci speller. *Intell. Neuroscience*, 2010, January 2010.

[4] O Chapelle, B Schölkopf, and A Zien. *Semi-supervised learning*. MIT Press, Cambridge, MA, 2006.

[5] E. Maby, G. Gibert, P.E. Aguera, M. Perrin, O Bertrand, and J. Mattout. The openvibe p300-speller scenario: a thorough online evaluation. In *Human Brain Mapping Conference 2010*, Barcelona, Spain, 2010.

[6] LR Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *IEEE proc.*, 77(2):257–286, 1989.

[7] Pieter-Jan Kindermans, Hannes Verschore, David Verstraeten, and Benjamin Schrauwen. A p300 bci for the masses: Prior information enables instant unsupervised spelling. In *Advances in Neural Information Processing Systems (NIPS 2012), year = 2012*.

[8] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.