An Adaptive Ensemble Model for Brain-Computer Interfaces

Isao Hayashi
Department of Informatics, Kansai University
Takatsuki, Osaka 569–1095, Japan
Email: ihaya@cbii.kutc.kansai-u.ac.jp

Shinji Tsuruse
Panasonic System Solutions Japan Co.,Ltd.
Chuo Ward, Tokyo 104–0061, Japan
Email: suraimuppoi@yahoo.co.jp

***Abstract*—Brain-computer interface (BCI) have recently entered the research limelight. In many such systems, external computers and machines are controlled by brain activity signals measured using near-infrared spectroscopy (NIRS) or electroencephalograph (EEG) devices. In this paper, we propose a probabilistic data interpolation-boosting algorithm for BCI, where we adopt three evaluation criterions to decide the class of interpolated data around the misclassified data. By using the interpolated data with classes, the discriminated boundary is shown to control the external machine effectively. We verify our boosting method with numerical examples, and discuss the results.**

***Index Terms*—Brain-Computer Interface, Boosting Algorithm, Probabilistic Data Interpolation.**

I. Introduction

We have proposed a boosting algorithm [1], [2] for brain computer interface [3], which interpolates data around misclassified data in order to extract the signals discriminated boundaries. We call our method pdi-Boosting (Probabilistic Data Interpolation-Boosting) [4]–[7]. In the BCIs, brain activity signals are measured using near infra-red spectroscopy (NIRS) [8]–[10] and electroencephalographic (EEG) devices [11]. Then, the external machine and computer are controlled by the models of the pdi-Boosting identified from brain activity signals. The pdi-Boosting consists of multiple weak classifiers, and the final output is determined by the result of a majority rule decision between the weak classifiers, done to improve overall discriminant accuracy. However, the external computer is sometimes unable to follow dynamic changes as the identified model uses prior brain activity data. Because, the interpolated data around the misclassified data are put as a class same as misclassified data. Thus, the class of interpolated data may not be necessarily put as the class same as the misclassified data.

In this paper, we propose a new class decision method to decide the class of interpolated data. The class of the interpolated data is not labeled by the misclassified data. Instead, three evaluation criterions, which are the evaluation of misclassified data, the evaluation of classification classes, and the evaluation of neighborhood classes, are defined, and the interpolated data is put as the class. The discriminant model of the enhanced pdi-Boosting is characterized by the addition to the class of new data generated around the misclassified data using the evaluation criterion. On the other hand, the discriminant curve is updated only by chosen individual data because Adaboost only updates the weight for data when the data set is constituted. A discriminant curve of pdi-Boosting draws a smoother trace by the difference of the number of data which give the influence to the curve, and the recognition rate of pdi-Boosting is improved. Therefore, as the enhanced pdi-Boosting generates interpolated data with the new class around the misclassified data, we can obtain a discriminant boundary with inherent robustness, and the external computer is able to follow dynamic changes in the environment. We evaluate the enhanced pdi-Boosting for BCI by numerical examples. We show usefulness of pdi-Boosting by the numerical example which is easy issue to classify to two classes daringly to clarify characteristic difference between Adaboost and pdi-Boosting. First, we explain the pdi-Boosting algorithm and discuss the characteristics of pdi-Boosting by simple numerical examples where we add various amounts of disturbances. Second, we propose the enhanced pdi-Boosting algorithm and show the usefulness of our method by simple numerical examples.

II. Formulation of pdi-Boosting

AdaBoost [12], [13] is an outstanding boosting method. In each iteration we select training data (TRD) from the set of misclassified data with high weights of over 50%, and then apply this data to a weak classifier in the consecutive iteration. After identifications are made by the weak classifier, the weights of the data are updated. After iterating the procedure sequentially, the final output is determined by majority rule decision of the multiple weak classifiers $M_1, M_2, \cdots, M_i, \cdots, M_L$, when the checking data (CHD) is given to these models.

In the pdi-Boosting algorithm, new data is interpolated around the misclassified data using a probability density function instead of the updating of the weights as in AdaBoost. A conceptual diagram of pdi-Boosting is shown in Fig. 1; the algorithm is as follows. Similar to AdaBoost, the final output is determined by a rule of majority decision using the multiple weak classifiers, when the checking data (CHD) is given to these models. However, the difference between AdaBoost and pdi-Boosting is that the amount of data in pdi-Boosting increases as compared to AdaBoost as shown in Fig. 2.

Assume that the misclassified data is given as the d-th data in TRD, and the j-th attribute of the d-th data is denoted

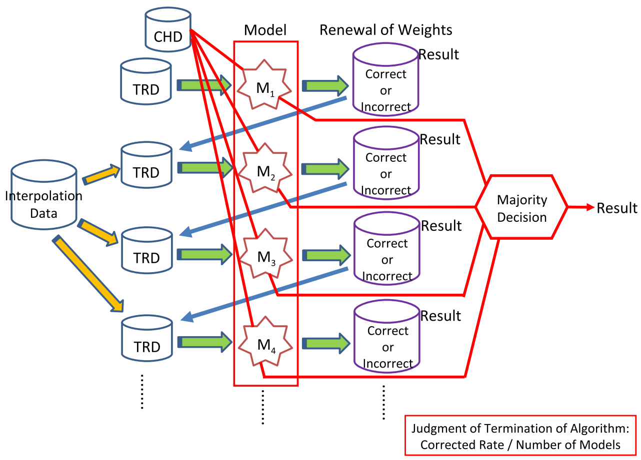

Fig. 1. Conceptual Diagram of pdi-Boosting

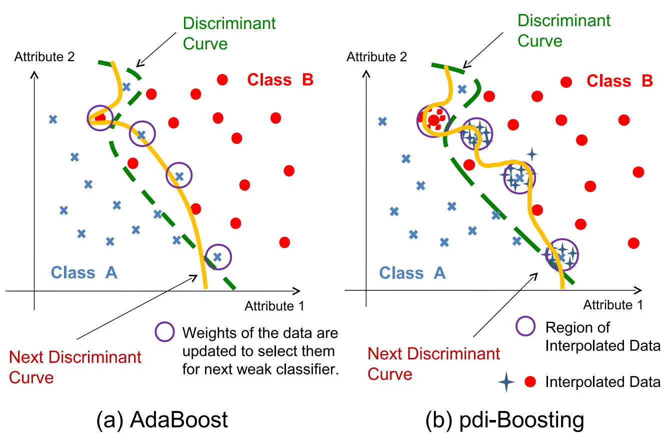

Fig. 2. Conceptual Difference between AdaBoost and pdi-Boosting

by $x_j^F(d)$. The interpolated data $x_j^{int}(d)$ is generated by a probability density function $f(x_j)$ around the misclassified data $x_j^F(d)$ with mean value.

$$x_j^{int}(d) = \{x_j \in A \mid P(A) = \int_A f(x_j)dx_j\} \quad (1)$$

In general we choose the normal distribution function to be our probability density function $f(x_j)$ with standard deviation σ as follows:

$$f(x_j) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(-\frac{(x_j - x_j(d))^2}{2\sigma^2}). \quad (2)$$

However, we may also adopt a uniform distribution as follows:

$$f(x_j) = \begin{cases} \frac{1}{x_j^{max} - x_j^{min}} & ; \quad for\ x_j^{min} \leq x_j \leq x_j^{max} \\ 0 & ; \quad for\ x_j < x_j^{min}\ or\ x_j > x_j^{max} \end{cases} \quad (3)$$

where x_j^{max} and x_j^{min} are defined as

$$x_j^{min} = \frac{3x_j(d) + x_j(d-1)}{4} \quad (4)$$

$$x_j^{max} = \frac{3x_j(d) + x_j(d+1)}{4} \quad (5)$$

and $x_j(d-1)$ and $x_j(d+1)$ denote the $(d-1)$-th and the $(d+1)$-th data, respectively.

We formulate the algorithm of pdi-Boosting as follows:

Step 1 The brain activity data D of size W is divided to two data sets: the training data D^{TRD} with the size W^{TRD}, and the checking data D^{CHD} with the size W^{CHD}, where $W = W^{TRD} + W^{CHD}$. In addition, the interpolated data from D is denoted by D^{INT}.

Step 2 The training data D^{TRD} is given as input into the i-th weak classifier M_i. The recognition rate r_i^{TRD} is calculated and the result given as R_i.

Step 3 The d-th misclassified data is selected from D^{TRD}. With this d-th data, a new interpolated data $x_j^{int}(d)$ is generated around $x_j^F(d)$ of the j-th attribute by the probability density function $f(x_j)$ defined in equations (2) through (5), and this new data $x_j^{int}(d)$ is saved to D^{INT}.

Step 4 Interpolated data are extracted from D^{INT} until the number of misclassified data are the same as the number of correctly classified data, where the number of interpolated data v satisfies

$$v \geq \frac{W}{2} - W^{TRD}(1 - r_i^{TRD}). \quad (6)$$

Step 5 Let θ be the threshold value and K is the number of iterations. The algorithm terminates when either one of the conditions $K = i$, $r_i^{CHD} \geq \theta$ or $i \geq K$ is satisfied.

Step 6 We apply D^{CHD} to $M_1, M_2, \cdots, M_i, \cdots, M_K$ to obtain the final discriminant result with recognition rate r_i^{CHD}.

Because new data is added around the misclassified data using a probability density function, and in each successive iteration the weak classifier fits the misclassified data closer than in the previous iteration, the final result will more closely approximate the given data.

III. Evaluation of pdi-Boosting

We simulate examples of signals using numerical data to discuss the evaluation on pdi-Boosting. We assume that oxyhemoglobin (oxy-Hb) and deoxyhemoglobin (doxy-Hb) data are normalized to lie in the range $[0, 1]$, where the number of clusters is 2. We assume the value corresponding to the steady state to be 0, and that to the activation state to be 1. We add five types of noise to the signals consisting of normally distributed random numbers of 500 with standard deviation s as follows:

Number of data: $2, 5, 10, 20, 30, 50, 75, 100, 250, 350, 500$
Standard deviation $\sigma = 0.005, 0.01, 0.05, 0.1, 0.2, 0.6, 1.0$
Standard deviation $s = 0.2, 0.4, 0.6, 0.8, 1.0$

We choose the normal distribution function to be the probability density function $f(x_j)$, and we adopted REPtree as the weak classifier, which is a type of decision tree method. The termination rule for the algorithm is set to be at iteration number $K = 3$. An evaluation of pdi-Boosting is shown as the following results.

1) The recognition rate does not decrease even though the number of data is extremely small. In addition, when the number of the data is large, the recognition rate is high and its variance is small.
2) The recognition rate in case of the learning data is high even if the number of data is few.
3) When noise is large, the robustness of the recognition rate is high.

As the first result, the recognition rate of pdi-Boosting was higher than AdaBoost even though the number of data is extremely less than ten. In addition, when the number of the data is large, the recognition rate of pdi-Boosting was higher than AdaBoost. As the second result, the recognition rate of pdi-Boosting in case of the learning data was higher than AdaBoost even if the number of data is few. As the third result, the larger the noise, the higher the recognition rate of pdi-Boosting compared with AdaBoost.

We must pay in particular attention to the third result. The recognition rate of pdi-Boosting and AdaBoost is shown in Fig. 3 when the standard deviation s for generating noise was changed with 0.4, 0.8, and 1.0. In Fig. 3, the recognition rate of pdi-Boosting is drawn in solid line, and AdaBoost is drawn in dashed line. The cognitive rate is the mean value of ten times of trials. The larger the noise, the lower the recognition rate of pdi-Boosting and AdaBoost. The variance of recognition rate is extremely large in the case of the number of data less than 50. However, the recognition rate of pdi-Boosting is higher than AdaBoost, when the number of data is over 100. In addition, the recognition rate of pdi-Boosting increases in proportion to the number of data. From these results, pdi-Boosting has a robustness as to noise compared with AdaBoost. As a result, we must notice that the recognition rate of pdi-Boosting is higher than AdaBoost.

We discuss the recognition rate when the standard deviation σ for generating probability distribution function $f(X_j)$ was changed. When we assume a position of misclassified data at 0.5, we show the frequency distribution of the interpolation data in the figure 4. The larger the standard deviation σ, the wider the domain of interpolation data. In the case of $\sigma = 1.0$, around 80% of data are included in interval of $[0, 1]$. Therefore, we notice that we must define larger standard deviation, when more extensive interpolation data are necessary.

IV. Enhanced pdi-Boosting

The interpolated data around the misclassified data were put as a class same as misclassified data. However, the class

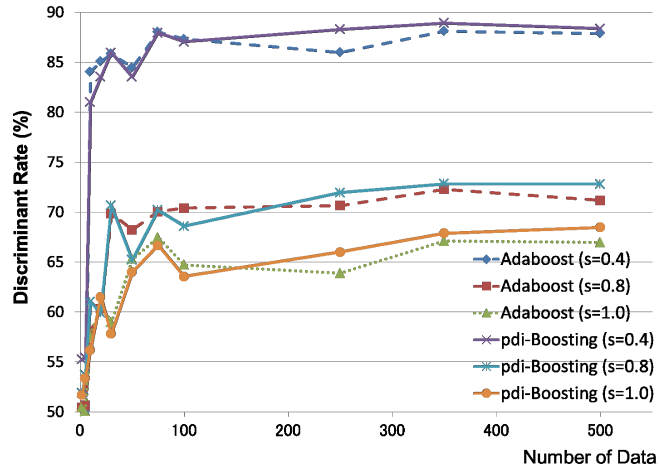

Fig. 3. Discriminant Rate by Changing Number of Data and S.D.

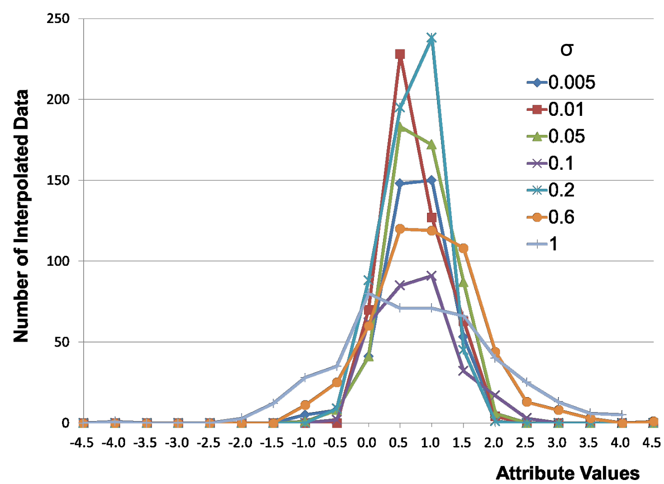

Fig. 4. Frequency Distribution of Interpolated Data

of interpolated data may not be necessarily put as the class same as the misclassified data. Therefore we propose a new class decision method to decide the class of interpolated data. Assume that the interpolated data $x_j^{int}(d)$ is generated from the misclassified data $x_j^F(d)$. Three evaluation criterions, which are the evaluation of misclassified data E_1, the evaluation of classification classes E_2, and the evaluation of neighborhood classes E_3, is defined, and the interpolated data $x_j^{int}(d)$ is put as a class k^*.

(1) Evaluation of Misclassified Data

Evaluation E_{j1} is defined by the probability distribution function $f(x_j)$, and that represents the dependence of the interpolated data to the misclassified data (See in Fig. 5). Evaluation E_{j1} shows that the dependence of the interpolated data to the misclassified data is high when E_{j1} of the interpolated data is large.

$$E_{j1}^k = \begin{cases} P(x_j^{int}(d)) & x_j^F \in k \\ 1 - P(x_j^{int}(d)) & x_j^F \notin k \end{cases}$$

$$P(x_j^{int}(d)) = \int_{x_j^F(d)}^{x_j^{int}(d)} f(x_j) dx_j$$

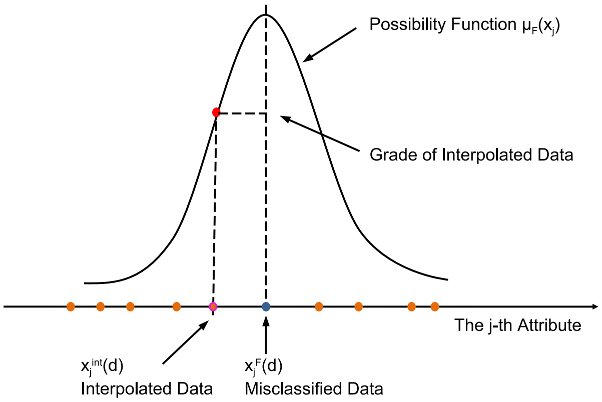

Fig. 5. Evaluation E1

(2) Evaluation of Classification Classes

Evaluation E_{j2} is defined by distance between the interpolated data and the center of each classification class (See in Fig. 6). Evaluation E_{j2} shows that the dependence of the interpolated data to the classification class is high when E_{j2} of the interpolated data is small.

$$E_{j2}^k = \frac{|x_j^{int}(d) - x_c^k| - min_i|x_i^k - x_c^k|}{max_i|x_i^k - x_c^k| - min_i|x_i^k - x_c^k|}, \quad for \ \forall i$$

where, x_c^k is the center of the classification class k.

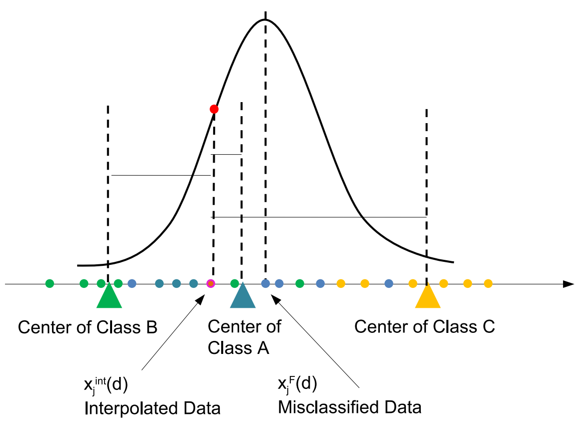

Fig. 6. Evaluation E2

(3) Evaluation of Neighborhood Classes

Evaluation E_{j3} is defined by distance to data x_j^N which is the nearest to the interpolated data in each classification class (See in Fig. 7). Evaluation E_{j3} shows that the dependence of the interpolated data to the neighborhood class is high when E_{j3} of the interpolated data is small.

$$E_{j3}^k = \frac{|x_j^N - x_j^{int}(d)| - min_i|x_i^k - x_j^{int}(d)|}{max_i|x_i^k - x_j^{int}(d)| - min_i|x_i^k - x_j^{int}(d)|}$$

$$for \ \forall i$$

The evaluation E_1 is large when interpolated data is generated

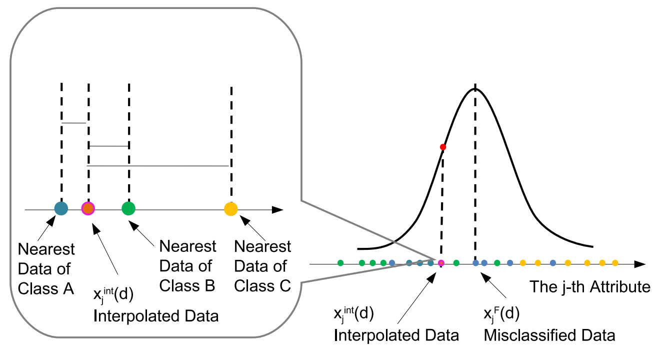

Fig. 7. Evaluation E3

around the misclassified data, and the evaluation E_2 is large when interpolated data is generated around the center of the identification class. On the other hand, the evaluation of the class distribution around interpolated data is calculated in evaluation E_3.

Evaluation E_j^k is calculated by the weighted summation of these three evaluation in each j attribute. The class of interpolated data $x_j^{int}(d)$ is defined as k^* where the total evaluation E^k of n attributes is minimized.

$$k^* = \{k | \min_k E^k = \min_k \sum_{j=1}^{n} E_j^k\} \quad (7)$$

$$E_j^k = w_1 E_{j1}^k + w_2 E_{j2}^k + w_3 E_{j3}^k \quad (8)$$

where, w_1, w_2, w_3 is weight at each evaluation, respectively.

We formulate the algorithm of enhanced pdi-Boosting as follows:

Step 1 The brain activity data D of size W is divided to two data sets: the training data D^{TRD} with the size W^{TRD}, and the checking data D^{CHD} with the size W^{CHD}, where $W = W^{TRD} + W^{CHD}$. In addition, the interpolated data from D is denoted by D^{INT}.

Step 2 The training data D^{TRD} is given as input into the i-th weak classifier M_i. The recognition rate r_i^{TRD} is calculated and the result given as R_i.

Step 3 The d-th misclassified data is selected from D^{TRD}. With this d-th data, a new interpolated data $x_j^{int}(d)$ is generated around $x_j^F(d)$ of the j-th attribute by the probability density function $f(x_j)$ defined in equations (2) through (5).

Step 4 The class k^* of the interpolated data $x_j^{int}(d)$ is distinguished by equations (7) and (8). This new data $x_j^{int}(d)$ is saved to D^{INT}.

Step 5 Interpolated data are extracted from D^{INT} until the number of misclassified data are the same as the number of correctly classified data, where the number of interpolated data v satisfies.

Step 6 Let θ be the threshold value and K is the number of iterations. The algorithm terminates when either one of the conditions $K = i$, $r_i^{CHD} \geq \theta$ or $i \geq K$ is satisfied.

Step 7 We apply D^{CHD} to $M_1, M_2, \cdots, M_i, \cdots, M_K$ to obtain the final discriminant result with recognition rate r_i^{CHD}.

V. Evaluation of Enhanced pdi-Boosting

We simulate examples of signals as two-cluster problem using numerical data to discuss the evaluation on enhanced pdi-Boosting. We assume that brain activity signals of 490 are normalized to lie in the range $[0, 1]$ and $[-1, 0]$, and the value of the data corresponding to the steady state to be 0, and that to the activation state to be 1 or -1. We add four types of noise to the signals consisting of normally distributed random numbers with standard deviation s. We choose the normal distribution function to be the probability density function, and we adopted REPtree as the weak classifier, which is a type of decision tree method. The termination rule for the algorithm is set to be at iteration number $K = 3$.

The four patterns are given in Fig. 8 to 11, respectively.

Weights: $w_1 = w_2 = w_3 = 1/3$
Standard deviation $\sigma = 0.0001$
Standard deviation $s = 0.2, 0.4, 0.6, 0.8$

We compare enhanced pdi-Boosting with normal pdi-Boosting, REPtree and other conventional Boosting algorithms, and we discuss the recognition rates at the different standard deviation. The comparison results are summarized in Table I. We show that the average recognition rate is 10 times that of the disturbance data. In Table I, the recognition rate of enhanced pdi-Boosting is shown to be slightly higher than that of other methods with standard deviation s ranging from 0.2 to 0.8. The comparison of enhanced pdi-Boosting and normal pdi-Boosting, shows that the recognition rate of enhanced pdi-Boosting is only 1.28% higher than that of normal pdi-Boosting on average for all four patterns. In addition, comparing enhanced pdi-Boosting with the other methods, AdaBoost, MultiBoost and REPTree, the recognition rate of enhanced pdi-Boosting is only 0.58% higher than

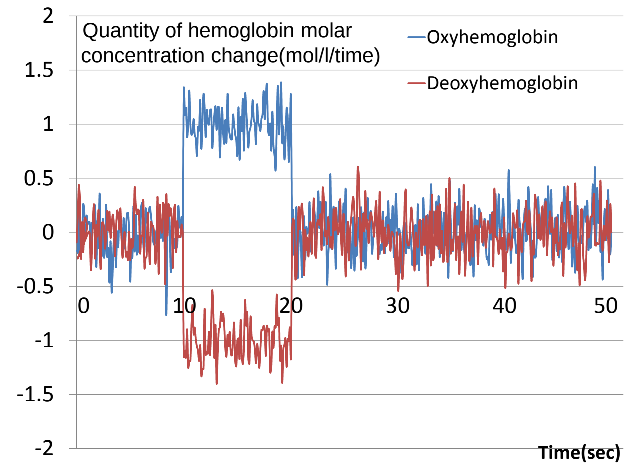

Fig. 8. Data of s=0.2

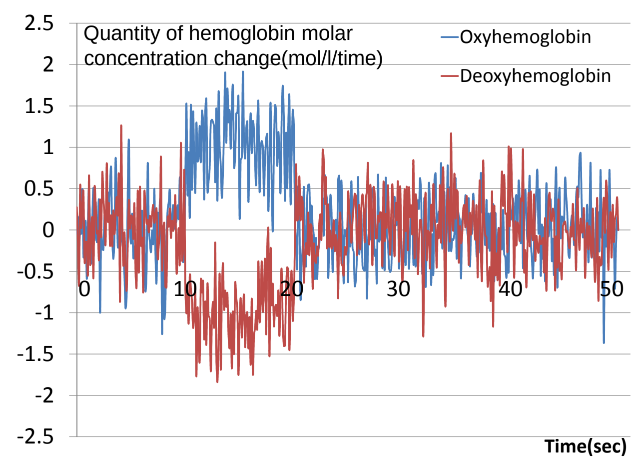

Fig. 9. Data of s=0.4

that of AdaBoost, 1.05% higher than that of MultiBoost and 2.53% higher than that of REPTree on average for all four patterns. Unfortunately, the recognition rate of enhanced pdi-Boosting couldn't show a significant difference as compared with other boosting methods by the multiple comparison of tukey method. However, the recognition rate of enhanced pdi-Boosting shows a significant difference ($p = 0.000037$) as compared with normal pdi-Boosting by the t-test with significance level 0.01%. Looking at these results, enhanced pdi-Boosting is more accurate than other boosting methods even though the significance of enhanced pdi-Boosting is not clear by a test.

At the stage $K = 1$ of the standard deviation $s = 0.8$, the number of all data became 866. Since the number of original data is 490, 376 as difference data are the interpolated data. The interpolated data which changed the class by new evaluation algorithm set off 67.0% of the total data. In those data, the interpolation data which were changed from the steady state "zero" to the activation state "one" were 73, and the interpolated data changed the steady state with the

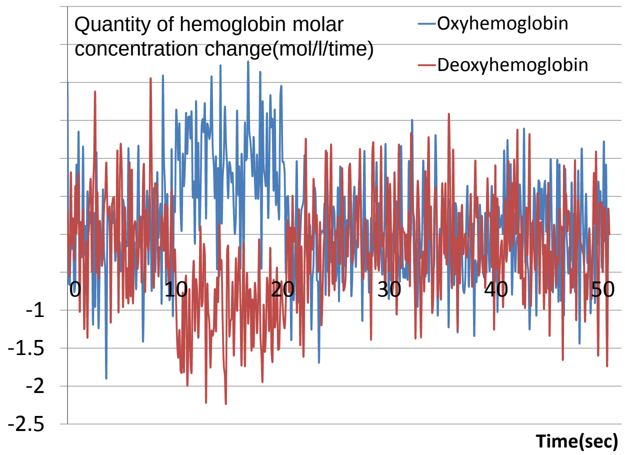

Fig. 10. Data of s=0.6

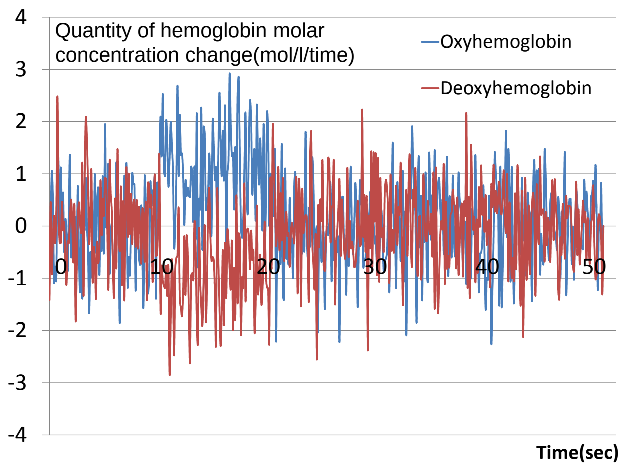

Fig. 11. Data of s=0.8

activation state were 181. We should notice that the recognition rate of pdi-Boosting is having improved by the change of these classes more. Therefore, we conclude that the use of the proposed pdi-Boosting algorithm is advantageous in practical BCI applications.

TABLE I
COMPARISON OF ENHANCED PDI-BOOSTING WITH OTHER METHODS

SD	Enh. pdi-B. (%)	pdi-B. (Uniform) (%)	Ada Boost (%)	Multi Boost (%)	REP Tree (%)
0.2	99.81	99.63	99.80	99.76	99.37
0.4	97.32	96.18	96.57	94.84	97.14
0.6	92.65	90.05	91.22	91.22	92.41
0.8	88.78	87.69	88.63	88.55	88.33
Ave.	94.64	93.38	94.06	93.59	92.11

VI. CONCLUSION

In this paper, we discussed our classification method based on a boosting algorithm using a probabilistic data interpolation scheme. In addition, we verified our method with numerical examples, and discussed the effectiveness of our new approach by comparing its performance to that of conventional boosting algorithms. In future work, we plan to discuss how to optimize the probability density functions used, and how to apply this method to a range of practical BCI problems.

ACKNOWLEDGMENT

This work was supported in part by the Ministry of Education, Culture, Sports, Science, and Technology of Japan under Strategic Project to Support the Formation of Research Bases at Private Universities 2008-2012, in addition the Organization for Research and Development of Innovative Science and Technology (ORDIST) of Kansai University.

REFERENCES

[1] Y.Freund: "Boosting a Weak Learning Algorithm by Majority," *Information and Computation*, Vol.121, No.2, pp.256-285 (1995).
[2] T.Nakashima, and Y.Shoji: "The Effect of Data Partition in Constructing Fuzzy Ensemble Classifiers," *Proceedings of the 25th Fuzzy System Symposium*, No.3E2-01 (2009) (in Japanese).
[3] M.A.Lebedev, J.M.Carmera, J.E.O'Doherty, M.Zacksenhouse, C.S.Henriquez, J.C.Principe, and M.A.L.Nicolelis, "Cortical ensemble adaptation to represent velocity of an artificial actuator controlled by a brain-machine interface," *Journal of Neuroscience*, Vol.25, No.19, pp.4681-4693 (2005).
[4] I.Hayashi, and S.Tsuruse: "A Proposal of Boosting Algorithm for Brain-Computer Interface Using Probabilistic Data Interpolation," *IEICE Technical Report, Neurocomputing*, Vol.109, No.461, pp.303-308 (2010) (in Japanese).
[5] I.Hayashi, and S.Tsuruse: "A Proposal of Boosting Algorithm by Probabilistic Data Interpolation for Brain-Computer Interface," *Proceedings of the 26th Fuzzy System Symposium*, pp.288-291 (2010) (in Japanese).
[6] I.Hayashi, S.Tsuruse, J.Suzuki, and R.T.Kozma: "A Proposal for Applying pdi-Boosting to Brain-Computer Interfaces," *Proceedings of 2012 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE2012) in 2012 IEEE World Congress on Computational Intelligence (WCCI2012)*, pp.635-640 (2012).
[7] I.Hayashi, and S.Tsuruse: "An Evaluation of pdi-Boosting for Brain-Computer Interfaces," *Proceedings of the 6th International Conference on Soft Computing and Intelligent Systems and the 13th International Symposium on Advanced Intelligent Systems (SCIS-ISIS2012)*, pp.1215-1220 (2012).
[8] M.Wolf, G.Morren, D.Haensse, T.Karen, U.Wolf, J.C.Fauchere, and H.U.Bucher: "Near Infrared Spectroscopy to Study the Brain: An Overview," *Opto-Electronics Review*, Vol.16, No.4, pp.413-419 (2008).
[9] R.Sitaram, H.Zhang, C.Guan, M.Thulasidas, Y.Hoshi, A.Ishikawa, K.Shimizu, and N.Birbaumer: "Temporal Classification of Multichannel Near-Infrared Spectroscopy Signals of Motor Imagery for Developing a Brain-Computer Interface," *Neuroimage*, Vol.34, No.4, pp.1416-1427 (2007).
[10] W.Niide, T.Tsubone, and Y.Wada: "Discrimination of moving limb with near-infrared spectroscopy," *IEICE Technical Report, Neurocomputing*, Vol.107, No.542, pp.191-196 (2008) (in Japanese).
[11] T.Yamaguchi, K.Nagata, Q.T.Pham, G.Pfurtscheller, and K.Inoue: "Pattern Recognition of EEG Signal during Motor Imagery by Using SOM," *International Journal of Innovative Computing, Information and Control*, Vol.4, No.10, pp.2617-2630 (2008)
[12] Y.Freund, and R.E.Schapire: "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *Journal of Computer and System Sciences*, Vol.55, No.1, pp.119-139 (1997).
[13] R.E.Schapire: "The Boosting Approach to Machine Learning: An Overview," In D.D.Denison, M.H.Hansen, C.Holmes, B.Mallick, and B.Yu, editors: *Nonlinear Estimation and Classification*, Springer (2003).