SENSITIVITY ANALYSIS OF BOOSTING PSI-BLAST WITH CASE STUDY ON SUBCELLULAR LOCALIZATION

F. Mai[1], *M.W. Mak*[2], *Y.S. Hung*[1], *S.Y. Kung*[3]

[1]Dept. of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong

[2]Center for Multimedia Signal Processing, Dept. of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong

[3]Dept. of Electrical Engineering, Princeton University, USA

ABSTRACT

This paper studies the sensitivity of PSI-BLAST with respect to the '*h*' parameter. Observing that the standard PSI-BLAST is sensitive to parameter '*h*' in the high-value region, we propose a new technique, called Boosting PSI-BLAST, to reduce the sensitivity. By constraining '*h*' to a small value first so as to reduce the chance of early corruption and then relaxing it gradually to increase divergence, the boosting PSI-BLAST not only can reduce the sensitivity to *h*-value, but also may strike a good balance between corruption and divergence in profiles. Tests on Reinhardt and Hubbard's eukaryotic protein dataset verify that our method is better in reducing the sensitivity of profile alignment scores to *h*-value than the standard PSI-BLAST.

1. INTRODUCTION

Detection of evolutionary relationship of proteins via database search has led to biologically important findings. The evolutionary information given by the searching results can aid the recognition of distant similarities and thus improve the accuracy of protein classification. PSI-BLAST is a widely used database searching program that is particularly good at finding distant relationships of proteins [1, 2]. The basic idea is to search a target database for sequences that share similarity with a query sequence and to generate a profile — position specific scoring matrix (PSSM) — from the multiple alignment of the similar sequences. This searching process is repeated by using the PSSM in the last round as query in the current round until no new sequences are found, or the user specified maximum number of iterations is reached, whichever comes first [3]. There are two key parameters that control the searching process. One is the *j*-value that specifies the maximum number of iterations, and the other is the *h*-value (expectation value threshold), which is the statistical significance threshold for selecting matches from database sequences to construct the PSSM. Since a lower expectation value means a more significant alignment score, setting a lower *h*-value will make the search more stringent, so that PSI-BLAST will select fewer matches that give high alignment scores entirely by chance.

Previous researches have studied the properties of PSI-BLAST, finding that usually increasing the *h*-value or *j*-value will include more family members in creating the protein profile. This is good until a "chance", which results in profile "corruption" [4], as opposed to a "real" similarity is included . Schaffer *et al.* [3] investigated over a dozen modifications to PSI-BLAST, with the goal of improving the accuracy of finding true positive matches. They tested on a Saccharomyces cerevisiae and the nematode worm Caenorhabditis elegans dataset, and studied the PSSM corruption problem caused by improper use of *h*-value. They claim that "one can avoid most corruption by lowering '*h*' sufficiently, but one pays a price in search accuracy for the majority of queries that do not get corrupted." Przybylski *et al.* studied on the behavior of PSI-BLAST in protein secondary structure prediction [5]. They conclude that higher divergence should theoretically yield better prediction and that for protein secondary structure prediction, it is beneficial to include more distant homologues in the alignment even if some of them are false positives. The work on subcellular localization prediction also shows the advantage of high divergence [6]. The catch is that while we include more family members by relaxing the parameters of PSI-BLAST to increase the divergence, the risk of introducing more false positives (causing profile corruption) will increase too. Indeed, we also observe that the alignment score is sensitive to the *h*-value in the high *h*-value region.

To highlight how the alignment scores affect the prediction accuracy, we show three profile alignment score matrices obtained from Reinhardt and Hubbard's dataset [7] in Figure 1, where the three sets of profiles are produced by

setting (a) $h \to 0$ and $e \to 0$ (can be viewed as sequence-based alignment matrix), (b) $h = 0.001$ and $j = 3$, and (c) $h = 10$ and $j = 30$. In these figures, the darker the dot, the higher the value of the alignment score. The figures show that when '*h*' is too small or too large, the block diagonal dominance in the alignment score matrix will become obscure. This weak block diagonal dominance will in turn lead to poor prediction accuracy, as confirmed by our experimental results: the overall accuracy using PairProSVM [6] for case (a) is 88%, for case (b) is 99% and for case (c) is 70%. This performance variation is partially due to the sensitivity of the profile alignment scores with respect to *h*-values. [1]

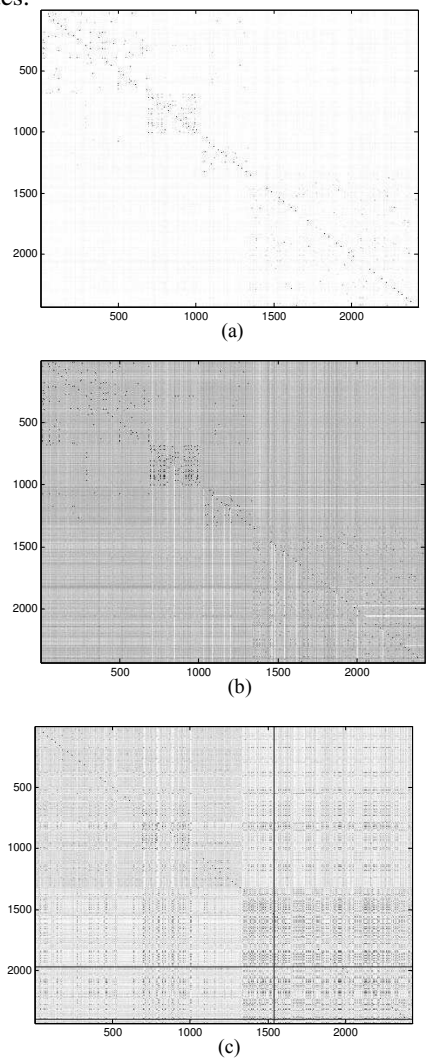


Fig. 1. Pairwise profile alignment score matrix (a) $h \to 0$ and $e \to 0$; (b) $h = 0.001$ and $j = 3$; (c) $h = 10$ and $j = 30$

It has been observed by us (for example, see Figures 5 to 7) and many authors that the alignment scores become sensitive to *h* when *h* far exceeds certain default value (10^{-3}). In particular, the sensitivity becomes severe when $h > 1$. Nevertheless, there are many reasons justifying the use of *h* in such a high-value range, one of which is for the secondary structure prediction [5]. Another application that may be benefited from using high h-value is the hierarchical classification of proteins. Note that classification of proteins can be divided into different levels, giving rise to a hierarchical structure shown in Figure 2. The h-value can be linked to the remoteness of homologous proteins, which in turn is related to the level of classification. Figures 1(b) and 1(c) show that the difference in alignment scores between classes becomes smaller when h increases from a moderate to a very large value. This suggests that increasing h may cause some of the classes to merge, resulting in a higher level of classification.

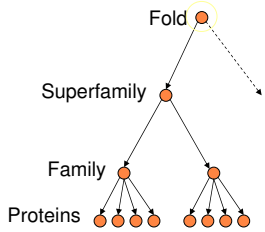


Fig. 2. Hierarchical classification of proteins

To reduce the risk of having profile corruption in the high h-value region, it is necessary to look into the effects of *h*-values. This paper proposes a boosting technique to reduce the sensitivity of PSI-BLAST. Sensitivity analysis on Reinhardt and Hubbard's eukaryotic protein dataset demonstrates that the boosting technique can reduce the sensitivity, especially in the high *h*-value region.

2. BOOSTING PSI-BLAST VS. STANDARD PSI-BLAST

2.1. Boosting Schedule

Serving as a black box, PSI-BLAST starts with a single query protein sequence as input and produces the corresponding profile as output, hiding the intermediate steps of multiple alignment and profile construction to users. Usually users are prompted to specify the parameters '*h*' and '*j*', which are mainly responsible for the divergence of the family found by PSI-BLAST. Increasing either the *h*-value or the *j*-value in PSI-BLAST will include more members into the family to generate the profile, but it may also introduce more false positives into the family, corrupting the profile and making the profile alignment score very sensitive to *h*-value. If false positives enter the list of matches at one iteration, they may corrupt the PSSMs in the subsequent iterations. The earlier the false positives are erroneously included into the family, the greater the corruption. *We propose to reduce the risk of early corruption by tightening the*

[1] In this paper, the term "sensitivity" refers to the amount of change in the alignment scores or the searching behavior of PSI-BLAST with respect to the change in the PSI-BLAST parameters such as the h-value.

h-value and j-value, and then relaxing them gradually to include more true family members. Based on this idea, we propose our boosting PSI-BLAST as follows:
Given a query protein sequence *A*

1. Define a pre-specified schedule consisting of a sequence of monotonic increasing *h*-values h_i, and a sequence of *j*-values $j_i (i = 0, 1, 2, \ldots, n)$.

2. Initialization: for $i = 0, h = h_0$ and $j = j_0$, use the standard PSI-BLAST to search against a database using Sequence *A* as query to generate a profile PSSM_0.

3. For $i = 1$ to n, iteratively apply the standard PSI-BLAST to search the database using PSSM_{i-1} as query for j_i iterations (or until convergence) with $h = h_i$ and generate profile PSSM_i.

4. Output PSSM_n as the profile of query sequence *A*.

This process is illustrated in Figure 3, where (a) describes the flow of boosting PSI-BLAST and (b) illustrates the *h*-value schedule needed in the boosting algorithm.

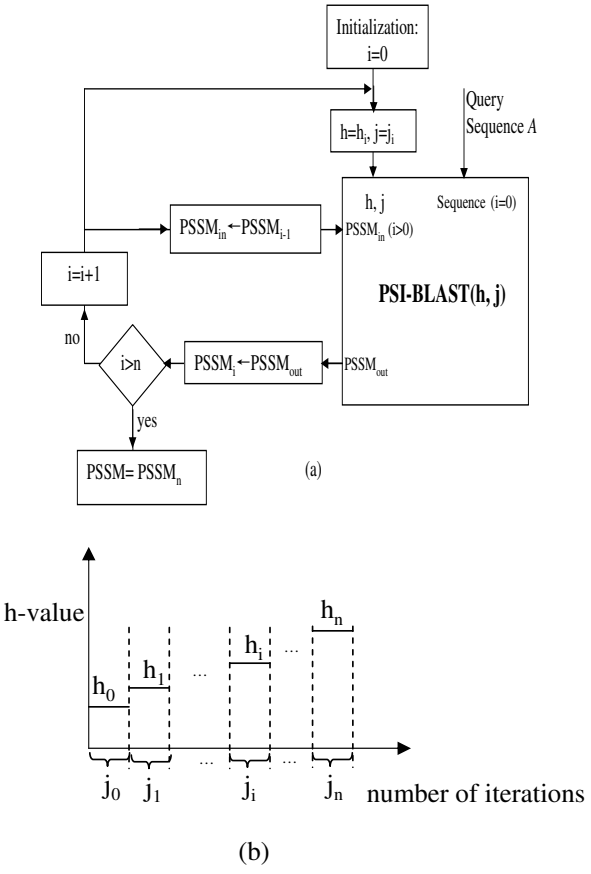


Fig. 3. (a) The flowchart of boosting PSI-BLAST; (b) *h*-value schedule in boosting PSI-BLAST (total number of iterations is $\sum_{i=1}^{n} j_i$)

The major difference between our boosting PSI-BLAST and the standard PSI-BLAST lies in the predefined schedule for *h* and *j*. In the standard PSI-BLAST, the *h*-value does not change during the iterations. In our boosting technique, the *h*-value in the *h*-schedule is chosen to have a small initial value and then relaxed in a monotonically increasing manner, i.e., $h_0 < h_1 < \ldots < h_i < \ldots < h_n$, as illustrated in Figure 3(b).

3. EXPERIMENTS AND RESULTS

3.1. Dataset and profile alignment scores

Reinhardt and Hubbard's dataset [7] which has been used extensively for evaluating subcellular localization methods in the literature [7, 8] was employed to analyse the behavior of the boosting PSI-BLAST. The sequences in this database were extracted from SWISSPORT 33.0 and the subcellular location of every protein has been annotated. The sequences were filtered, i.e., only those that appeared to be complete and having reliable annotations were kept. Transmembrane proteins and plant sequences were excluded. The resulting dataset comprises 2427 eukaryotic proteins (684 cytoplasm, 325 extracellular, 321 mitochondrial, and 1097 nuclear proteins). Following [6], we used local pairwise profile alignment scores to construct SVM kernels for predicting the subcellular locations of proteins. Because SVM kernels have significant influence on the prediction performance of SVM classifiers, we analyzed the sensitivity of alignment scores with respect to *h*-value. Through these analysis, we will be able to assess the ability of our method in reducing the sensitivity of PSI-BLAST with respect to *h*-value.

3.2. Selections of class representatives

Although large-scale simulation with statistical results will be our ultimate objective, in this paper we report our study based on representative pairwise alignment scores to provide insight into the issues involved. We select the representatives in each class by computing the ratio of intra-class alignment score to the inter-class alignment score. A sequence with a high score ratio should have high alignment scores with a large number of members in its own class, but low alignment scores with members in other classes. A sequence with a low score ratio should have similar alignment scores, either high or low, with all the other sequences, no matter in the same class or not. In Euclidean space, a centroid of a cluster should be the one that is closest to all the others. We borrow this centroid concept by defining a centroid in a class as a sequence with a high score ratio. Similarly, a sequence with a low score ratio is defined as a loner in a class. We found that in each class there are more than one controid, so we select several centroids and loners randomly as representatives. Figures 4(a) and 4(b) show an ex-

ample of one centroid and one loner in Class 1 (Cytoplasm) respectively.

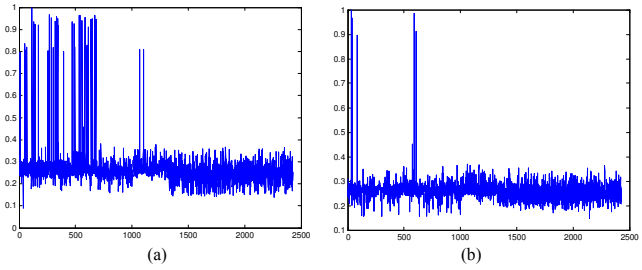


Fig. 4. Two representatives of Class1: (a) centroid and (b) loner. The horizontal axis represents the sequence indexes of four classes. *Class 1*:1–684; *Class 2*: 685–1009; *Class 3*: 1010–1330; *Class 4*: 1331–2427. The vertical axis represents the alignment scores of the selected representative with all the 2427 sequences

We conducted two groups of experiments: intra-class alignment, where both sequences to be aligned come from the same class, and inter-class alignment, where the sequences to be aligned come from two different classes. Due to space limitation, we will only give several examples to show the general trend.

3.3. Experimental details

For each pair of sequences, we generated the profiles by using the standard PSI-BLAST [2] and the boosting PSI-BLAST to search against the SWISSPROT 46.0 [9] for comparison. We selected 28 h-values ranging from 10^{-60} to 100: $h = [10^{-60}, 10^{-55}, 10^{-50}, 10^{-45}, 10^{-40}, 10^{-36}, 10^{-32}, 10^{-28}, 10^{-24}, 10^{-20}, 10^{-18}, 10^{-16}, 10^{-14}, 10^{-12}, 10^{-10}, 10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{0}, 10^{1/4}, 10^{1/2}, 10^{3/4}, 10^{1}, 10^{5/4}, 10^{3/2}, 10^{7/4}, 10^{2}]$. Thus protein sequence A has 28 profiles $A^k (k = 1, 2, \ldots, 28)$ with respect to the 28 h-values, and so does sequence B. Then we computed the profile alignment scores [6] $\zeta(A^k, B^k) \quad k = 1, 2, \ldots, 28$.

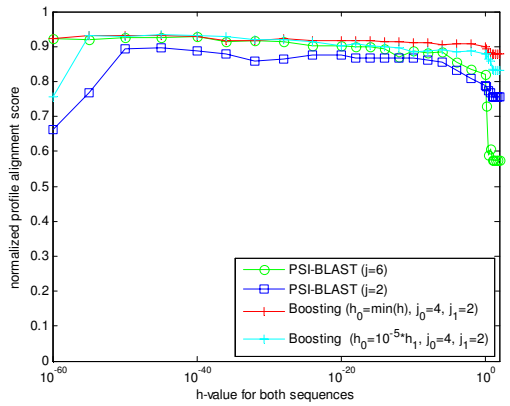
When creating the profiles using the original PSI-BLAST, we set $j = 6$ because it is suggested that running PSI-BLAST for five to six rounds may be enough to find most of the matches [3, 4]. However, we also tried $j = 2$ to see the influence of the j-value on the profile alignment.

When implementing the boosting PSI-BLAST, we chose an h scheme in which only h_0 and h_1 are defined. Accordingly, the j-values are set as $j_0 = 4$ and $j_1 = 2$ so that the total number of iterations is also 6. We tried both a conservative scheme and a greedy scheme. For the conservative scheme, we start the algorithm with the smallest h-value, i.e., $h_0 = 10^{-60}$, and then jump to the preset h-value h_1. For the greedy scheme, we set $h_0 = 10^{-5} h_1$.

3.4. Results on Reinhardt & Hubbard's dataset

For intra-class, we show 3 pairs in Class 3: centroid vs. centroid within the same cluster (Figures 5(a) and 5(b)), centroid vs. centroid in two different clusters and centroid vs. loner (Figures 6(a) and 6(b)). For inter-class cases, we show the alignment of one centroid in Class 3 with both the centroid and loner (Figures 7(a) and 7(b)) in Class 1.

(a)



(b)

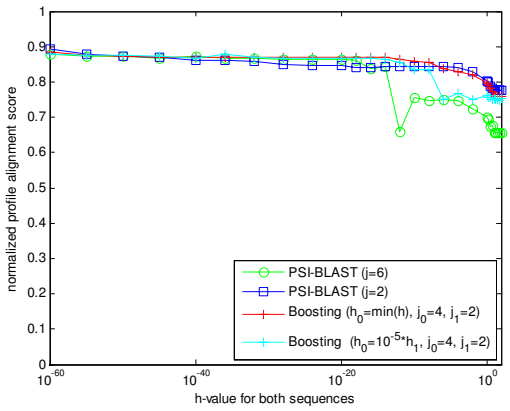


Fig. 5. Sensitivity analysis of intra-class sequence pairs (both are centroids) with high alignment score. (a) 'ACDL_HUMAN' vs. 'ACDM_HUMAN'; (b) 'HEM1_RAT' vs. 'HEM0_MOUSE'

3.5. Discussions

Before we look into the sensitivity results shown in Figures 5 to 7, let us apply winner-takes-all to the alignment score matrix shown in Figure 1 (b) in order to study the properties of high and low alignment scores. Figure 8 (a) shows the results, where the horizontal and vertical axes represent the sequence indexes of eleven class. *Class 1*: 1–684; *Class 2*: 685–1009; *Class 3*: 1010–1330; *Class 4*: 1331–2427. For each column, the entry (winner) with the highest score is represented by a black dot. Therefore, dots that appear in the block diagonal regions of the figure represent correct classifications. By computing the ratio of the

(a)

(b)

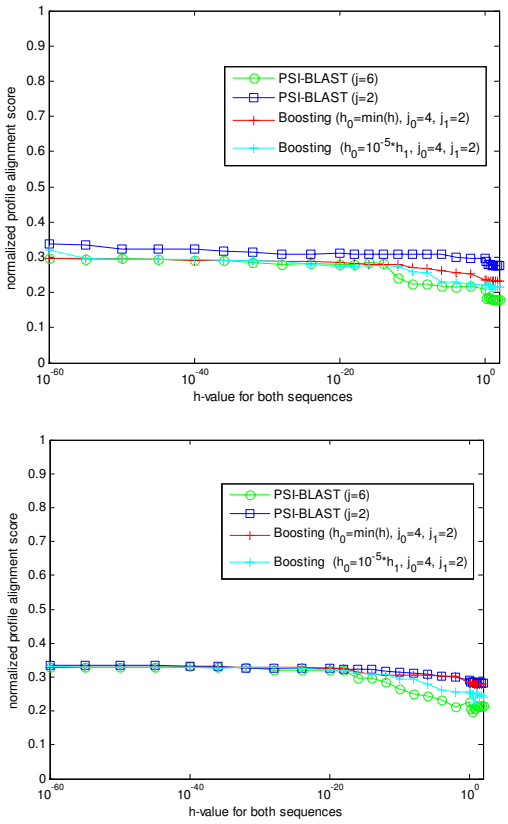


Fig. 6. Sensitivity analysis of intra-class sequence pairs with low alignment scores
(a) centroid vs. centroid: 'ACDL_HUMAN' vs. 'HEM1_CHICK'
(b) centroid vs. loner: 'ACDL_HUMAN' vs. 'BCS1_YEAST'

(a)

(b)

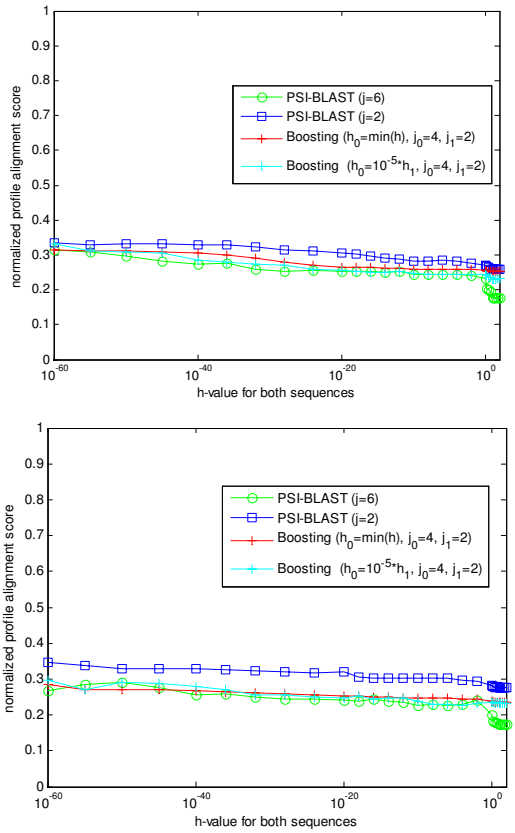


Fig. 7. Sensitivity analysis of inter-class sequence pairs
(a) centroid vs. centroid: 'ACDL_HUMAN' vs. 'AMY2_SALTY'
(b) centroid vs. loner: 'ACDL_HUMAN' vs. 'EFTU_NEIGO'

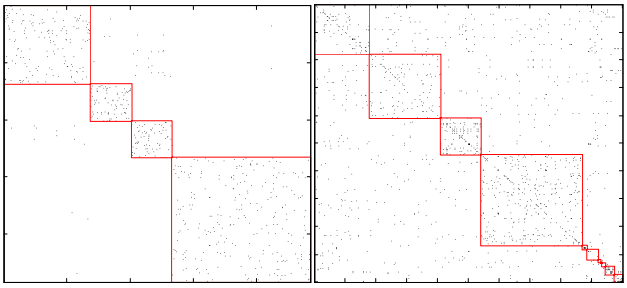
the number of correct classifications to the number of all sequences, we obtain a classification accuracy of 82%. Similar block diagonal structure also appears in Huang and Li's dataset [8] with a classification accuracy of 61%, as shown in Figure 8(b). This suggests that high alignment scores are more trustworthy than the low alignment scores, because by using only the high alignment scores, we could achieve the accuracy which is much better than the accuracy by chance ($\approx 25\%$). This point is also supported by Baldi [10]:"below the threshold some pairs will be related and some will not, so subthreshold matches cannot be used to obtain negative conclusion. It is known that proteins can be structurally very similar even if the sequence similarity is very low. At such low similarity levels, pure chance will produce other pairwise alignments that will mix with those produces by genuinely related pairs". This property of alignment scores together with Figures 5 to 7 allow us to (1) confirm and validate known results found by other researches and (2) bring new insight into the PSI-BLAST and boosting PSI-BLAST operating at high h-value region.

(1) Confirmation and Validation of Known Findings

(a) At low-to-medium values of 'h', the alignment scores produced by our boosting PSI-BLAST with $h_0 = \min(h)$, $j_0 = 4$, $j_1 = 2$ and the standard PSI-BLAST with $j = 6$ (red curves and green curves in Figures 5 to 7) exhibit similar sensitivity characteristics with respect to h-value, confirming the finding in [5] that alignment scores are not sensitive in the low-to-medium h-value.

(b) A comparison between the green curves (standard PSI-BLAST with $j = 6$) and blue curves (standard PSI-BLAST with $j = 2$) in Figures 5 to 7 reveals that the number of iterations in PSI-BLAST (j-value) is detrimental to the sensitivity of the alignment score, especially at high h-value region. This phenomenon is also confirmed in [5].

(2) New Insights and Findings Let us compare the red curves (Boosting PSI-BLAST with $h_0 = \min(h)$, $j_0 = 4$, $j_1 = 2$) and the green curves (standard PSI-BLAST with $j = 6$) in Figures 5 to 7. To provide a fair comparison, we set the total number of iterations for these two curves to be



(a) (b)

Fig. 8. Figure illustrating the classification performance of the winner-takes-all method on (a) R&H dataset and (b) H&L dataset. The horizontal and vertical axes represent the sequence indexes of the classes. For each column, the entry (winner) with the highest score is represented by a dot. Therefore, dots that appear in the block diagonal regions of the figure represent correct classifications.

identical ($= 6$), which allows us to focus on the effect of the h-value on the alignment scores.

(a) To achieve high prediction accuracy, it is desirable to have high alignment scores for intra-class cases (Figure 5) and low scores for inter-class cases (Figure 7). We note that the difference between the two curves in the high alignment-score case is larger than that in the low alignment-score case, especially in the high h-value region. In other words, the Boosting PSI-BLAST can make the high alignment scores higher than the PSI-BLAST, and at the same time, the low alignment scores not much worse than the standard PSI-BLAST.

(b) Now let us take a closer look at the high alignment score case. Note that the red curves produced by the boosting PSI-BLAST retain a reasonable stability. In contrast, the green curves produced by the standard PSI-BLAST exhibit very abrupt drop in the score, indicating undesirable corruption in the profile. Recall from our earlier discussion in Section 3.5 that it is more critical to maintain the high alignment scores in the intra-class case than reducing the low alignment scores in the inter-class case. Therefore, the boosting PSI-BLAST holds a critical advantage in providing a more systematic expansion of family size.

4. CONCLUSIONS

This paper proposes a boosting method, a new way of using PSI-BLAST, to reduce sensitivity of profile alignment scores to h-value. This is achieved by tightening the h-value first and relaxing it gradually. We have conducted sensitivity analysis on Reinhardt and Hubbard's eukaryotic protein dataset and found that careful choice of both the h-value and j-value is very important when using PSI-BLAST. By observing the pairwise profile alignment scores, we found that in the low-to-medium h-value range, our boosting method and the standard PSI-BLAST have similar sensitivity properties. In the high h-value range, our boosting method outperforms the standard PSI-BLAST in terms of the ability to reduce the sensitivity. This indicates that by using the boosting method, we may reach a better balance between increasing the divergency and reducing the chance of profile corruption. Since high sensitivity of profile alignment scores to h-value may deteriorate the prediction accuracy, we expect our method for reducing the h-value sensitivity can help improve the protein subcellular localization prediction accuracy.

Boosting PSI-BLAST allows enlarging the membership in the profile without suffering from severe corruption which is more likely to occur in PSI-BLAST. Therefore, by using boosting PSI-BLAST, some proteins from distinctive classes in the lower level may be more safely reclassified to the same class in the higher level. We advocate that the boosting PSI-BLAST is a valuable alternative for biologists to further study high-level classification and high-level protein family.

5. REFERENCES

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *J. Mol. Biol*, vol. 215, pp. 403–410, 1990.

[2] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped BLAST and PSI-BLAST: A new generation of protein database search programs," *Nucleic Acids Res.*, vol. 25, pp. 3389–3402, 1997.

[3] A. A. Schaffer, L. Aravind, T. L. Madden, S. Shavirin, J. L. Spouge, Y. I. Wolf, E. V. Koonin, and S. F. Altschul, "Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements," *Nucleic Acids Res.*, vol. 29, pp. 2994–3005, 2001.

[4] S. F. Altschul, private communication.

[5] A. Przybylski and B. Rost, "Alignments grow, secondary structure prediction improves," *Proteins.*, vol. 46, pp. 197–207, 2002.

[6] J. Guo, M.W. Mak, and S.Y. Kungl, "Eukaryotic protein subcellular localization based on local pairwise profile alignment SVM," in *IEEE International Workshop on Machine Learning for Signal Processing*, Maynooth, Ireland, 2006, pp. 391–396.

[7] A. Reinhardt and T. Hubbard, "Using neural networks for prediction of the subcellular location of proteins," *Nucleic Acids Res.*, vol. 26, pp. 2230–2236, 1998.

[8] Y. Huang and Y. D. Li, "Prediction of protein subcellular locations using fuzzy K-NN method," *Bioinformatics*, vol. 20, no. 1, pp. 21–28, 2004.

[9] B. Boeckmann, A. Bairoch, R. Apweiler, M. C. Blatter, A. Estreicher, E. Gasteiger, M. J. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, and M. Schneider, "The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003," *Nucleic Acids Res.*, vol. 31, pp. 365–370, 2003.

[10] P. Baldi and S. Brunak, *Bioinformatics : The Machine Learning Approach*, MIT Press, 1998.