BIOINFORMATICS ORIGINAL PAPER
Vol. 21 no. 11 2005, pages 2667–2673
doi:10.1093/bioinformatics/bti384

Structural bioinformatics

Fold recognition by combining profile–profile alignment and support vector machine

Sangjo Han†, Byung-chul Lee†, Seung Taek Yu, Chan-seok Jeong, Soyoung Lee and Dongsup Kim*

Department of Biosystems, Korea Advanced Institute of Science and Technology, Daejeon, 305-701, Korea

Received on December 28, 2004; revised on March 4, 2005; accepted on March 8, 2005
Advance Access publication March 15, 2005

ABSTRACT

Motivation: Currently, the most accurate fold-recognition method is to perform profile–profile alignments and estimate the statistical significances of those alignments by calculating *Z*-score or *E*-value. Although this scheme is reliable in recognizing relatively close homologs related at the family level, it has difficulty in finding the remote homologs that are related at the superfamily or fold level.

Results: In this paper, we present an alternative method to estimate the significance of the alignments. The alignment between a query protein and a template of length n in the fold library is transformed into a feature vector of length $n+1$, which is then evaluated by support vector machine (SVM). The output from SVM is converted to a posterior probability that a query sequence is related to a template, given SVM output. Results show that a new method shows significantly better performance than PSI-BLAST and profile–profile alignment with *Z*-score scheme. While PSI-BLAST and *Z*-score scheme detect 16 and 20% of superfamily-related proteins, respectively, at 90% specificity, a new method detects 46% of these proteins, resulting in more than 2-fold increase in sensitivity. More significantly, at the fold level, a new method can detect 14% of remotely related proteins at 90% specificity, a remarkable result considering the fact that the other methods can detect almost none at the same level of specificity.

Contact: kds@kaist.ac.kr

INTRODUCTION

Fold recognition is to recognize native-like structural folds of an unknown protein from the known protein structures. It provides not only the structural templates from which a detailed tertiary structure of a protein is predicted by comparative modeling, but also a means to increase our understanding of its biological function by detecting homologs that are difficult to detect by conventional homology search methods. It is also relevant to the target selection in the structural genomics initiatives (Kim, 1998) where one of the main goals is to experimentally determine enough protein structures to build the total repertoire of protein folds from which all proteins with unknown structure can be modeled by fold recognition and homology modeling (Friedberg *et al*., 2004; Hou *et al*., 2003a). In addition, by increasing the sensitivity of the fold-recognition method, we can increase the structural coverage of newly sequenced genomes (McGuffin *et al*., 2004).

In general, the fold recognition-methods fall into two classes. The first class uses solely the sequence information. The hidden Markov model (HMM) methods (Karplus *et al*., 1999), PSI-BLAST (Altschul *et al*., 1997), FFAS (Rychlewski *et al*., 2000), a method by Yona and Levitt (2002) and COMPASS (Sadreyev and Grishin, 2003) can be classified into this category. The second class uses the structural information in addition to the sequence information in various ways (Bowie *et al*., 1991). GenTHREADER (Jones, 1999), 3D-PSSM (Kelley *et al*., 2000), FUGUE (Shi *et al*., 2001), RAPTOR (Xu *et al*., 2003) and PROSPECT (Kim *et al*., 2003; Xu and Xu, 2000), to name a few, represent the fold-recognition methods that belong to the second class. It is known that including the evolutionary information for both the query and template proteins increases not only the fold-recognition performance but also the alignment quality (Kim *et al*., 2003; Ohlson *et al*., 2004). It is also known that the structural information, when it is used with the sequence information, increases the performance of the template-based tertiary structure prediction. Especially, the information on the predicted secondary structure of a query protein significantly improves the alignment quality (Elofsson, 2002) and fold-recognition performance (Przybylski and Rost, 2004). However, mounting evidence from the continuous benchmarking program of fold-recognition servers, such as LiveBench (Rychlewski *et al*., 2003) and the assessment on recent round of CASP (Kinch *et al*., 2003), suggests that the impact of the structural information is rather limited and the dominant factor in fold recognition is the quality of the profiles of both a query and a template.

Currently, a common strategy that the most accurate fold-recognition methods are adopting is to first perform sequence-profile (Altschul *et al*., 1997) or profile–profile (Wallner *et al*., 2004) alignments between a query sequence and the template sequences in the fold library, and then to estimate the statistical significances of those alignments by calculating *Z*-score (Kim *et al*., 2003; Shi *et al*., 2001) or *E*-value (Karlin and Altschul, 1990; Sadreyev and Grishin, 2003). Although this scheme is reliable in recognizing relatively close homologs related at the family level, reaching the sensitivity of ∼80% at 99% specificity, it still has difficulty in finding the remote homologs that are related at the superfamily or fold level, reaching only 25% sensitivity at 90% specificity at the superfamily level and almost zero sensitivity at the fold level (Ohlson *et al*., 2004). Here, we present an alternative way to estimate the

*To whom correspondence should be addressed.

†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

significance of the alignments by support vector machine (SVM) (Vapnik, 1998).

Utilizing the machine learning techniques, such as the artificial neural network (NN) or SVM, for the fold-recognition problem is not new. For example, GenTHREADER (Jones, 1999) and RAPTOR (Xu *et al*., 2003) transform query–template alignments to fixed length feature vectors, and then evaluate the feature vectors using NN and SVM, respectively, to produce a likelihood measure for each predicted fold recognition. The features used in these works include raw alignment score, alignment length, sequence lengths, pair energy evaluated by threading potential, salvation energy, sequence identity, etc. The critical difference between these previous works and the present method is that in this method all templates in template library have feature vectors of different lengths with profile–profile alignment scores at each position as their features, whereas the length of feature vectors for GenTHREADER and RAPTOR is the same for all templates. The methods, such as SVM-HMMSTER (Hou *et al*., 2004), SVM-I-sites (Hou *et al*., 2003b), SVM-pairwise (Liao and Noble, 2003) and SVM-Fisher (Jaakkola *et al*., 2000) are also closely related to the present method in many aspects in that these methods attempt to detect remote homologs by examining sequence alignments. In our method, the alignment between a query protein and a template of length n is transformed into a feature vector of length $n+1$ composed of n profile–profile alignment scores and a raw alignment score, $(sa^1, sa^2, \ldots, sa^i, \ldots, sa^n, \text{total\_score})$, where sa^i is the profile–profile alignment score at position i. Then, the feature vector is evaluated by SVM, and the output is converted to a posterior probability (Platt, 1999) that a query sequence is related to a template, given an SVM output. The test on large-scale benchmark set indicates that improvement over previous methods is quite dramatic; improvement is remarkable in the sensitivity of detecting remote homologs that are related at the superfamily and the fold levels.

MATERIALS AND METHODS

Data

We assess the fold-recognition performance of each algorithm by testing its ability to recognize related protein domains at the three different similarity levels, family, superfamily and fold, classified by the SCOP version 1.65 (Murzin *et al*., 1995). First, the fold library composed of ~5600 domains is constructed using domain subsets with <40% sequence identity to each other prepared by ASTRAL Compendium (Chandonia *et al*., 2004). We choose the folds containing at least 20 members for training the SVM and testing. A total of 62 folds and 2854 templates are selected as a result. Two-thirds of all templates in each fold (1885 templates) are randomly chosen and used for training the SVM for each template and the remaining one-third of templates (969 templates) are used for testing.

SVM feature vectors and training

To train SVMs for all 1885 templates in the training set, we first generate all-against-all alignments by profile–profile alignment scheme, without using any structural information. The profile–profile score to align the position i of a template q and the position j of a template t is given by $m_{ij} = \sum_{k=1}^{20} \left[f_{ik}^q S_{jk}^t + S_{ik}^q f_{jk}^t \right]$, where f_{ik}^q, f_{jk}^t, S_{ik}^q and S_{jk}^t are the frequencies and the position-specific score matrix (PSSM) scores of amino acid k, at position i of a template q and position j of a template t, respectively. The frequency matrices and PSSMs are generated by running PSI-BLAST using default parameters except for the number of iterations ($j = 6$). For each template of length n in the training set, alignments with the other 1884 templates in the training set are generated. Then, these 1884 alignments are transformed, respectively, into $(n+1)$-dimensional feature vectors, $(sa^1, sa^2, \ldots, sa^i, \ldots, sa^n, \text{total\_score})$, where sa^i is the profile–profile alignment score at position i of a given template and total_score the total profile–profile alignment score (Fig. 1). If gaps occur, fixed negative scores are arbitrarily assigned. Instead of using raw alignment scores, smoothened profile–profile alignment scores given by $sa^i = m_{i-2} + 2m_{i-1} + 3m_i + 2m_{i+1} + m_{i+2}$, where m_i denotes a raw profile–profile alignment score at position i of a template (Tress *et al*., 2003), are used. The total alignment scores are scaled to make their average and the standard deviation to equal 0 and 1, respectively, so that they have a comparable magnitude and range as the rest of the scores. Positive examples are the templates sharing the same fold with a target template for which an SVM is to be trained; otherwise, the templates are regarded as a negative example. In SVM training, the linear and radial basis function (RBF) kernels are tried, without attempting serious performance optimization. Freely available SVM software, svm_light (http://svmlight.joachims.org/), is used for SVM training and testing.

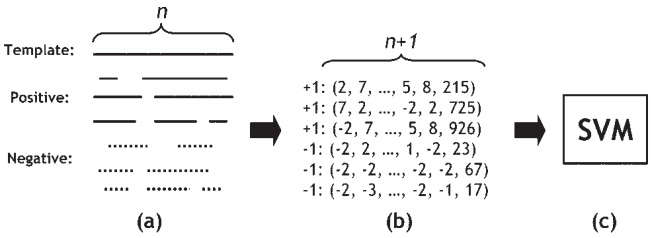

Fig. 1. Generation of the input feature vectors from alignments. (**a**) The sequence of a template of length n is aligned to the sequences of positive (solid line) and negative (dot line) examples by profile–profile alignment method. (**b**) Each alignment is transformed to $(n+1)$-dimensional feature vector composed of the alignment scores at n positions and the total alignment score. (**c**) These feature vectors, after scaling, are used to train SVM for a target template.

Testing and performance assessment

For each sequence in the testing set, the profile–profile alignments with all templates in the training set are generated and transformed to the feature vectors, which are then evaluated by the trained SVMs to produce outputs for all templates. A well-known problem of SVM is that the output values are neither calibrated nor directly associated with probabilistic meaning (Platt, 1999). The fact that a template α produces a greater SVM output than a template β does not imply that the alignment of a query protein with a template α is more significant than that with a template β. Therefore, ranking the templates by their raw SVM outputs is problematic. Instead, a posterior probability that a query sequence is related to a template, given an SVM output is estimated by the procedure proposed by Platt (1999). In Platt's method, an SVM output, f, and the posterior probability, $p(y = 1|f)$, are related by a sigmoid with two parameters, A and B; $p(y = 1|f) = 1/[1 + \exp(Af + B)]$. Ideally, the parameters, A and B, should be estimated from an independent set other than training and testing sets. However, owing to the scarcity of members in some folds, it is hard to form an independent set that has enough number of members. Therefore, we estimate the parameters as follows. For each template in the training set and a query protein, we pretend that the testing set has a query protein only and the independent set is the testing set minus a query protein. We estimate the parameters using the independent set and finally calculate the posterior probability. The same procedure is repeated for all proteins in the testing set. By doing so, we can avoid the possibility of introducing any bias. We have found that using the posterior probabilities instead of the SVM raw outputs improves the performance slightly. We measure fold-recognition performance of various methods in two different ways, the receiver operating characteristic (ROC) scores and the specificity–sensitivity plot. The ROC score is the area under the ROC curve, the plot of true positives as a function of the number of false positives (Gribskov and

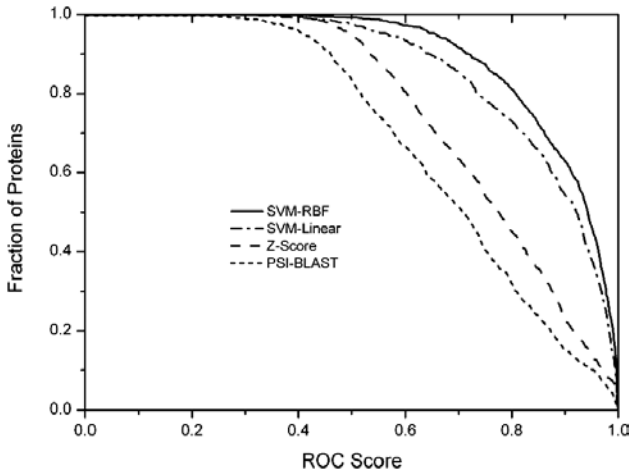

Fig. 2. ROC scores of various methods. The x-axis and y-axis represent the ROC score and the fraction of the proteins with a given performance, respectively. SVM-RBF, SVM-linear, Z-Score and PSI-BLAST denote SVM method with RBF kernel, SVM method with linear kernel, Z-score method and PSI-BLAST, respectively.

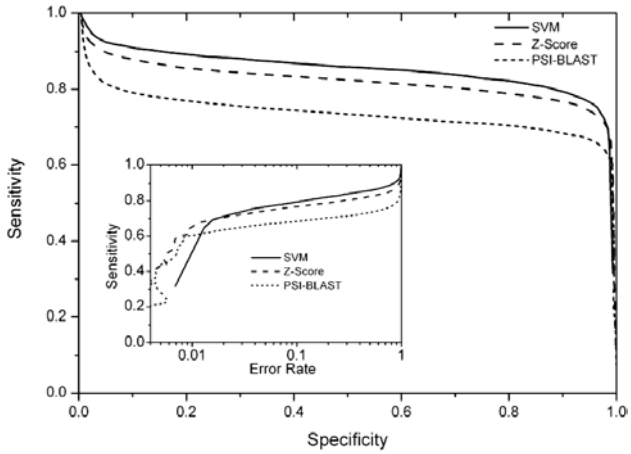

Fig. 3. The specificity–sensitivity plot for various methods at the family level. The inset plot shows the sensitivity as a function of error rate, which is defined by 1 − specificity. SVM, Z-Score, PSI-BLAST denote the SVM method with RBF kernel, Z-score method and PSI-BLAST, respectively. The sensitivity of the present method is better than those of the other two methods by ∼98% specificity. However, beyond 98% specificity, the performance of the SVM method deteriorates rather sharply compared with the other two methods.

Robinson, 1996). The highest score is 1, which indicates that all positives are ranking higher than all negatives. For randomly ordered list of positives and negatives, the score is expected to be 0.5, while a score <0.5 indicates that the ranking by a method is worse than random ordering. The Specificity is defined as Specificity $= \mathrm{TP}/(\mathrm{TP} + \mathrm{FP})$, where TP and FP denote the numbers of true and false positives, respectively, given a cutoff score. It measures the probability, that a pair with a score greater than a given cutoff score is a related protein pair at each similarity level. On the other hand, the Sensitivity is defined as Sensitivity $= \mathrm{TP}/(\mathrm{TP} + \mathrm{FN})$, where FN is the number of false negatives given a cutoff score. It is the fraction of the number of related proteins that are correctly recognized among all related proteins.

Running PSI-BLAST and calculating Z-scores

When running PSI-BLAST, we first generate the profiles for the sequences in the testing set using 'nr' database and the default parameter values, except for the number of iterations set to 10. Then using these profiles we search the database composed of the sequences in the training set. For each sequence in the testing set, the sequences in the training set are ranked according to their E-values. We have tried a few different options for parameters, such as E-value cutoff, the number of iterations etc. However, the results are more or less the same. To calculate the Z-score, a randomly shuffled sequence of a query protein is aligned to the template. By repeating the same process 100 times, the average and the standard deviation of the alignment score distribution, and eventually Z-score, are estimated.

RESULTS

In this section, we describe the fold-recognition performance of the present method, compared with those of PSI-BLAST method and the Z-score scheme.

In Figure 2, the ROC scores of various methods are shown. It is clear that the SVM method with the radial basis function (RBF) kernel significantly outperforms PSI-BLAST and the conventional Z-score scheme, demonstrating the superior performance of the present method. We have not tried the other SVM kernels, such as polynomial or sigmoid kernels, as other previous works on fold recognition (Xu *et al*., 2003) and the secondary structure prediction (Hua and Sun, 2001) using the SVM suggest that RBF kernel generally gives the best performance. Although we have not tried to systematically optimize the parameters, we believe that the current performance is not far from the optimum. From now on all the SVM results are obtained by using the RBF kernel. The observation that the profile–profile alignment with Z-score scheme outperforms PSI-BLAST is not surprising in that numerous previous studies (Kim *et al*., 2003; Ohlson *et al*., 2004; Sadreyev and Grishin, 2003; Von Ohsen *et al*., 2004) have demonstrated similar results. Comparing the improvement of the conventional Z-score scheme over PSI-BLAST and that of the present method over the conventional Z-score scheme, it is rather surprising how big a performance gain we can achieve by changing the method used to estimate the significance of the alignments while keeping the same profile–profile alignments. Using the SVM method with RBF kernel, 63% of all proteins in the testing set have >0.9 of ROC score, while the corresponding figures using Z-score scheme and PSI-BLAST are 23 and 15%, respectively. Moreover, with a new method, >20% of proteins achieve a near-perfect separation (ROC score of 0.99) of the positives from the negatives, while the corresponding figures were only 7 and 4% with Z-score scheme and PSI-BLAST, respectively. The major portion of performance improvement is attributable to the correct recognition of the remote homologs that are related to a query protein at the fold level, most of which PSI-BLAST and the conventional Z-score scheme typically fail to recognize. This is the direct consequence of training each template to recognize its remote homologs.

Figure 3 shows the specificity–sensitivity plot for the three methods at the family level, the SVM method with RBF kernel, the conventional Z-score method and PSI-BLAST. Also shown in the inset plot of Figure 3 is the sensitivity as a function of error rate, which is defined as 1 − specificity to show more clearly the performance of the three methods at a low error rate region. It is easy to recognize that the sensitivity of the new method is better than those of the other two methods by ∼98% specificity. However, beyond 98% specificity, the performance of a SVM method deteriorates rather sharply compared with the other two methods. This drop in performance at very low

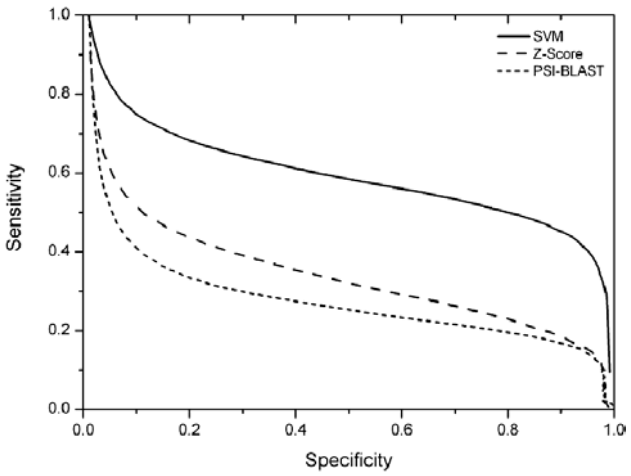

Fig. 4. The specificity–sensitivity plot for various methods at the superfamily level. SVM, *Z*-Score, PSI-BLAST denote the SVM method with RBF kernel, *Z*-score method and PSI-BLAST, respectively.

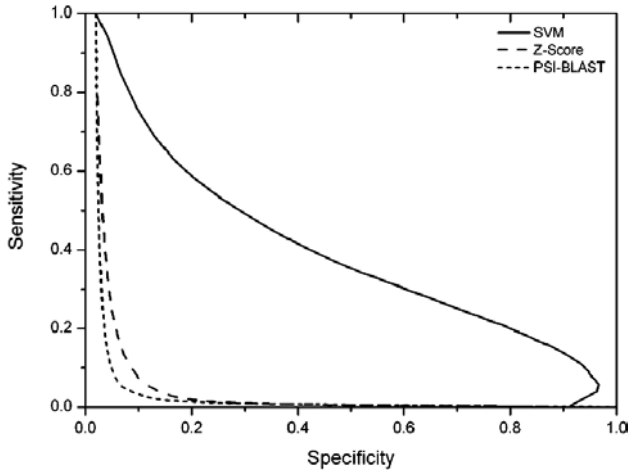

Fig. 5. The specificity–sensitivity plot for various methods at the fold level. SVM, *Z*-Score, PSI-BLAST denote the SVM method with RBF kernel, *Z*-score method and PSI-BLAST, respectively.

error rate (very high specificity region) is due to a few high-scoring false positives that are erroneously recognized by the SVM. The main reason for this problem is the definition of the positive examples with which the SVMs are trained. Since a protein that is related to the template at the fold level of similarity is considered as a positive example, and the alignment accuracy between the two proteins that are related at the fold level of similarity can be very low in some cases, the SVMs, that are trained based on those alignments, make a few obvious mistakes. However, this kind of problem is not serious because it can be dealt with in many ways. One way is to examine the scores of other templates in the fold library; if a certain high-scoring template is not related to all the other high-scoring templates that are related, or if most of the related templates of a certain high-scoring template all have low scores, it is likely that the high score of that template is the result of an error in the SVM method. In fact, a similar idea of using the global structure in the protein similarity network to improve homology search has been reported to be useful (Weston *et al*., 2004).

Performance improvement of the present method over the existing methods is more clearly shown in Figure 4 where a similar specificity–sensitivity plot at the superfamily level is shown. PSI-BLAST and *Z*-score scheme detect 16 and 20% of superfamily-related proteins at 90% specificity, respectively. According to a recent study on the fold-recognition performances of PSI-BLAST and the profile–profile alignment methods (Wallner *et al*., 2004), PSI-BLAST and the profile–profile methods detect 16 and 20% of superfamily-related proteins at 90% specificity, respectively. Although they used a different benchmark set, their results are nearly the same as our calculation, which implies that difficulty in benchmarking our testing set is nearly the same as with that of their set. Meanwhile, the present method detects 46% of these proteins, resulting in more than 2-fold increase in sensitivity compared with the previous methods.

More significantly, as shown in Figure 5, at 90% specificity the present method can detect as much as 14% of remotely related proteins at the fold level, whereas the other methods can detect almost none at the same level of specificity. This result has important implications in many different but related aspects. One aspect is the target selection (Brenner, 2000) in the structural genomics initiatives (Kim, 1998). A recent analysis on TargetDB (Chen *et al*., 2004), a database for the target proteins of all structural genomics centers, suggests that the target proteins are highly redundant and the percentage of novel folds among solved structures is not as high as expected (Bourne *et al*., 2004). To achieve the original goal set by structural genomics initiatives, which is to experimentally determine enough protein structures to build the total repertoire of protein folds from which all proteins with unknown structure can be modeled using fold recognition and homology modeling, it is important to select as a target, the proteins that are most likely to have a novel fold. Thanks to the high sensitivity of the present method, failure to recognize a likely fold of a candidate protein will increase the possibility that a protein may have a novel fold. The other aspect is the structural annotation coverage of a genome. According to a recent study (McGuffin *et al*., 2004), on average 64% of the proteins encoded in a genome can be confidently assigned to known folds. The main reason for such low coverage rate is the existing fold-recognition program's inability to detect remote homologs. It is expected that by improving fold-recognition method we can significantly increase the structural coverage rate, thereby increasing the capability of structural annotation of a new genome. The biggest implication is on the template-based protein structure prediction. The quality of predicted protein structure depends on choosing the best structural template and the alignment accuracy. Moreover, the applicability of the template-based protein structure prediction methods is limited by our ability to recognize a correct fold. As is evident from Figure 5, many existing best-performing fold-recognition methods fail to recognize structural analogs that are related to a query protein at the fold level. By increasing the sensitivity of fold recognition at the fold level, it is possible to significantly increase the applicability of the template-based protein structure prediction methods, as well as the accuracy of protein structure prediction.

DISCUSSION

Why does the SVM method work so well? The reason for its success is related to the intermediate sequence search (Park *et al*., 1997)

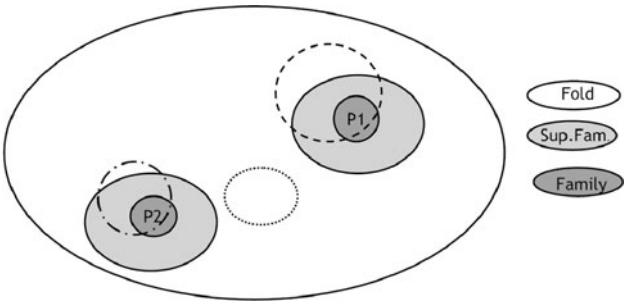

Fig. 6. Schematic illustration of remote homolog search by SVM. P1 can be related to proteins in the area enclosed by the dash line by typical homolog search algorithms. P1 can also recognize proteins in the area enclosed by the dash–dot line through an intermediate sequence P2. In addition, proteins in the area enclosed by dot line can be recognized by SVM's ability to learn the essential features of the fold.

and its ability to recognize the essential features among alignments of remotely related proteins. The situation is schematically depicted in Figure 6. In a situation where we try to search for all the proteins that are related to the template P1, by existing homology search algorithms we can typically find proteins in the area enclosed by the dash line; most family members, roughly a third of superfamily members and a few fold members. However, in the present method, P1 has been trained to recognize the sequences that are similar to P2. Therefore, it will also recognize proteins in the area enclosed by dash–dot line. This situation is similar to the intermediate sequence search (Park *et al*., 1997), where two homologous proteins are related through an intermediate sequence that are recognized to be homologous to the two sequences; P2 acts as an intermediate sequence. In addition, proteins in the area enclosed by dot line can be recognized by the present method, even though these proteins cannot be reached by an intermediate sequence search. Since we train SVM to learn some essential features of the alignments between proteins that belong to the same fold, the present method can detect remotely related proteins if the alignments between these proteins and P1 contain those essential features.

A good example is a domain 1kfwA2 (SCOP version d.26.3.1). In Table 1, the names and the scores of high-ranking templates for a domain 1kfwa2 using SVM method and Z-score scheme are shown. The top five templates (1goiA3, 1ll7A2, 1hjxA2, 1edqA3 and 1itxA2) share the same family with 1kfwA2, and are easily recognized by both methods, yielding high posterior probabilities and Z-scores by SVM method and Z-score scheme, respectively. The next template by SVM is 3eipA_ with a high probability of 0.84, which shares the same fold with a query, while the Z-score scheme fails to recognize the template 3eipA_ with a very low Z-score of -0.7. The reason that 3eipA_ is recognized by SVM but not by Z-score scheme is that when we train 3eipA_ using SVM, all top five templates (1goiA3, 1ll7A2, 1hjxA2, 1edqA3 and 1itxA2) are used as a positive example. As a result, 3eipA_ is trained to be recognized by proteins that are similar to these five templates, which is 1kfwA2, in this example. In fact, 1goiA3, 1ll7A2, 1hjxA2, 1edqA3 and 1itxA2 are all acting as an intermediate sequence that connects a query and 3eipA_.

In many situations, for instance, when we want to predict the structure of an unknown protein, the fact that a method can recognize a template from the same fold is not so significant, if templates from

Table 1. Top seven templates using SVM method with their probability and the ranking, along with the Z-score and the ranking by Z-score

Domain (SCOP version.)	Probability (rank)	Z-Score (rank)
1goiA3 (d.26.3.1)	1.0 (1)	18.0 (1)
1ll7A2 (d.26.3.1)	1.0 (1)	14.4 (2)
1hjxA2 (d.26.3.1)	1.0 (1)	13.1 (3)
1edqA3 (d.26.3.1)	1.0 (1)	12.0 (4)
1itxA2 (d.26.3.1)	1.0 (1)	11.5 (5)
3eipA_ (d.26.2.1)	0.84 (6)	−0.7 (1440)
1n1uA_ (g.3.3.1)	0.44 (7)	0.8 (459)

A query sequence is a domain 1kfwA2 (SCOP version d.26.3.1). Note that the top five templates share the same family with a query and the sixth template only shares the same fold.

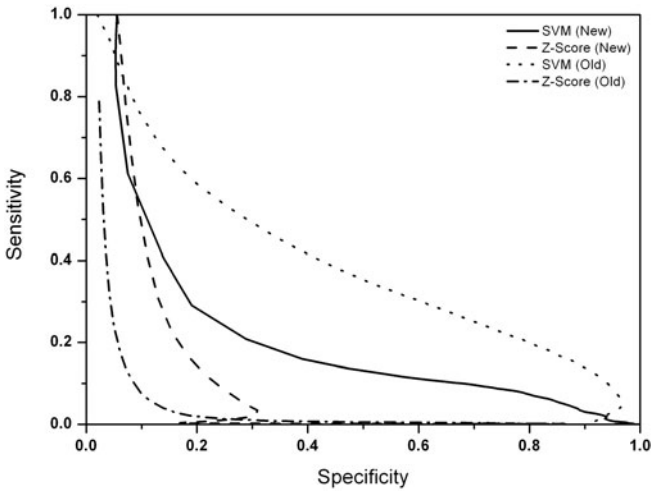

Fig. 7. The specificity–sensitivity plot for the new set and the old set at the fold level. SVM and Z-Score denote the SVM method and the Z-score method, respectively. In the new set, the training and testing sets share the proteins from the same superfamily. The old set denotes the training and testing sets described in the Materials and methods Section.

the same family are also recognized with higher scores, just like the above example, because that template will not be used for the structure prediction anyway. A difficult, but significant case is when there are only the templates that are similar to a query protein at the fold level. To assess the effectiveness of the present method in such a case, we prepare the training and testing sets that share no proteins from the same superfamily. The training set has 1288 templates and the testing set has 602 templates. This case is much more difficult than the previous one because there are no intermediate sequences from the same family and superfamily that connect the training and test sets. Only the sensitivity at the fold level can be discussed. The result is shown in Figure 7. Higher sensitivity for a new set at low specificity region is purely an artifact due to the fact that a new set has higher percentage of true positives than the previous set. It is clear that the present method outperforms the Z-score scheme significantly; the sensitivity of Z-score method drops to near-zero level at the specificity of $\sim$0.3, while the sensitivity of a new method remains at the significant level, up to the specificity of 0.8. Apparently, the

sensitivity of a new method for the new set is roughly half of that for the previous testing set.

Since this drop in sensitivity is due to the absence of intermediate sequences in the new set, it is reasonable to argue that it is mostly the method's ability to recognize the 'essential features' that are responsible for the present method's higher sensitivity for the new testing set, compared with the conventional Z-score scheme. As in most machine learning approaches, it is hard to pinpoint what features are important for the performance, therefore, it is not clear what the 'essential features' are. Nonetheless, we believe that the 'essential features' are from relatively small regions of alignments that are important, functionally and structurally, to a group of related proteins, and that the SVM can recognize these important alignment features. Let us suppose that an important region and a not-so-important region equally contribute to the total alignment score. Then, these two regions contribute equally to the PSI-BLAST's E-value and Z-score. On the other hand, the SVM can distinguish an important region from the not-so-important region and assign higher weight to the important region and ignore the other regions in calculating the posterior probability.

The method described so far is by no means optimized. The feature vectors used in the present work are relatively simple, and can be improved in many ways. We have tried the raw alignment scores instead of smoothened profile–profile alignment scores, but the performance seems to get slightly worse. We have found that by including the total scores in the feature vectors we can improve the performance slightly. When the total scores are included, the fraction of proteins having the ROC scores of 0.9 and 0.99 increases to 0.62 and 0.18 from 0.57 and 0.16, respectively. The current feature vectors do not include the structural features. It is expected that by including the structural features we may further improve the performance of the method. The limitation of the present method is that in order to properly train the SVM for a certain fold, a reasonable number of members should be in the fold. It is well known that the distribution of the number of members in folds follows the power law (Qian *et al*., 2001), which implies that a significant number of folds have only a few (non-redundant) proteins with known structures. In our template library with 5463 members, 3855 members belong to only 137 folds that have $\geq$10 members, and the remaining 1608 members are spread among 650 folds that have $<$10 members. One advantage of the power law distribution is that these 137 folds cover $\sim$70% (=3855/5463) of all proteins. Therefore, if the minimum number of members in a fold required for the present method to be effective is assumed to be 10, we may apply the present method to $\sim$70% of cases. We may further increase the applicability of the method by including remote homologs with unknown structure as a positive example. Of course, if these homologs are included, we can not use the structural features in the feature vectors. However, we believe that the importance of the structural features is rather limited.

In summary, in order to improve fold recognition, we develop a new method to estimate the significance of the alignments by SVM. The alignment between a query protein and a template of length n is transformed into a feature vector of length $n + 1$, and then this feature vector is evaluated by SVM. The output from an SVM is converted to a posterior probability that a query sequence is related to a template given an SVM output. Tests on benchmark set demonstrate that the new method show significantly better performance compared with not only PSI-BLAST but also the profile–profile alignment with Z-score scheme. Improvement is most prominent in recognizing remotely related proteins at the fold level. This high sensitivity at the fold level makes the present method a promising tool not only for the protein structure prediction but also the target selection in the structural genomics initiatives and increasing structural coverage of a genome.

ACKNOWLEDGEMENTS

This work is supported by CHUNG Moon Soul Center for BioInformaion and BioElectronics (CMSC). Computational resources are provided in part by the IBM SUR Grant.

REFERENCES

Altschul,S.F. *et al*. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Bourne,P.E. *et al*. (2004) The status of structural genomics defined through the analysis of current targets and structures. *Pac. Symp. Biocomput.*, 375–386.

Bowie,J.U. *et al*. (1991) A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, **253**, 164–170.

Brenner,S.E. (2000) Target selection for structural genomics. *Nat. Struct. Biol.*, **7** (Suppl), 967–969.

Chandonia,J.M. *et al*. (2004) The ASTRAL Compendium in 2004. *Nucleic Acids Res.*, **32** (Database issue), D189–D192.

Chen,L. *et al*. (2004) TargetDB: a target registration database for structural genomics projects. *Bioinformatics*, **20**, 2860–2862.

Elofsson,A. (2002) A study on protein sequence alignment quality. *Proteins*, **46**, 330–339.

Friedberg,I. *et al*. (2004) The interplay of fold recognition and experimental structure determination in structural genomics. *Curr. Opin. Struct. Biol.*, **14**, 307–312.

Gribskov,M. and Robinson,N.L. (1996) The use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. *Comput. Chem.*, **20**, 25–34.

Hou,J. *et al*. (2003a) A global representation of the protein fold space. *Proc. Natl Acad. Sci. USA*, **100**, 2386–2390.

Hou,Y. *et al*. (2003b) Efficient remote homology detection using local structure. *Bioinformatics*, **19**, 2294–2301.

Hou,Y. *et al*. (2004) Remote homolog detection using local sequence–structure correlations. *Proteins*, **57**, 518–530.

Hua,S. and Sun,Z. (2001) A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach. *J. Mol. Biol.*, **308**, 397–407.

Jaakkola,T. *et al*. (2000) A discriminative framework for detecting remote protein homologies. *J. Comput. Biol.*, **7**, 95–114.

Jones,D.T. (1999) GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. *J. Mol. Biol.*, **287**, 797–815.

Karlin,S. and Altschul,S.F. (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl Acad. Sci. USA*, **87**, 2264–2268.

Karplus,K. *et al*. (1999) Predicting protein structure using only sequence information. *Proteins*, (Suppl 3), 121–125.

Kelley,L.A. *et al*. (2000) Enhanced genome annotation using structural profiles in the program 3D-PSSM. *J. Mol. Biol.*, **299**, 499–520.

Kim,D. *et al*. (2003) PROSPECT II: protein structure prediction program for genome-scale applications. *Protein Eng.*, **16**, 641–650.

Kim,S.H. (1998) Shining a light on structural genomics. *Nat. Struct. Biol.*, **5** (Suppl), 643–645.

Kinch,L.N. *et al*. (2003) CASP5 assessment of fold recognition target predictions. *Proteins*, **53** (Suppl 6), 395–409.

Liao,L. and Noble,W.S. (2003) Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships. *J. Comput. Biol.*, **10**, 857–868.

McGuffin,L.J. *et al*. (2004) The genomic threading database: a comprehensive resource for structural annotations of the genomes from key organisms. *Nucleic Acids Res.*, **32** (Database issue), D196–D199.

Murzin,A.G. *et al*. (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Ohlson,T. *et al*. (2004) Profile–profile methods provide improved fold-recognition: a study of different profile–profile alignment methods. *Proteins*, **57**, 188–197.

Park,J. *et al*. (1997) Intermediate sequences increase the detection of homology between sequences. *J. Mol. Biol.*, **273**, 349–354.

Platt,J.C. (1999) Probabilistic outputs for support vector machines and comparisons to regularized likehood methods. In Smola,A., Bartlett,P., Scholkopf,D. and Schuurmans,D. (eds), *Advances in Large Margin Classifiers*. MIT Press, Cambridge, MA, pp. 61–74.

Przybylski,D. and Rost,B. (2004) Improving fold recognition without folds. *J. Mol. Biol.*, **341**, 255–269.

Qian,J. *et al*. (2001) Protein family and fold occurrence in genomes: power-law behaviour and evolutionary model. *J. Mol. Biol.*, **313**, 673–681.

Rychlewski,L. *et al*. (2000) Comparison of sequence profiles. Strategies for structural predictions using sequence information. *Protein Sci.*, **9**, 232–241.

Rychlewski,L. *et al*. (2003) LiveBench-6: large-scale automated evaluation of protein structure prediction servers. *Proteins*, **53** (Suppl 6), 542–547.

Sadreyev,R. and Grishin,N. (2003) COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance. *J. Mol. Biol.*, **326**, 317–336.

Shi,J. *et al*. (2001) FUGUE: sequence–structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. *J. Mol. Biol.*, **310**, 243–257.

Tress,M.L. *et al*. (2003) Predicting reliable regions in protein alignments from sequence profiles. *J. Mol. Biol.*, **330**, 705–718.

Vapnik,V. (1998) *Statistical Learning Theory*. Wiley, New York.

Von Ohsen,N. *et al*. (2004) Arby: automatic protein structure prediction using profile–profile alignment and confidence measures. *Bioinformatics*, **20**, 2228–2235.

Wallner,B. *et al*. (2004) Using evolutionary information for the query and target improves fold recognition. *Proteins*, **54**, 342–350.

Weston,J. *et al*. (2004) Protein ranking: from local to global structure in the protein similarity network. *Proc. Natl Acad. Sci. USA*, **101**, 6559–6563.

Xu,J. *et al*. (2003) RAPTOR: optimal protein threading by linear programming. *J. Bioinform. Comp. Biol.*, **1**, 95–117.

Xu,Y. and Xu,D. (2000) Protein threading using PROSPECT: design and evaluation. *Proteins*, **40**, 343–354.

Yona,G. and Levitt,M. (2002) Within the twilight zone: a sensitive profile–profile comparison tool based on information theory. *J. Mol. Biol.*, **315**, 1257–1275.