Clique Scoring for Enhancing Protein Family Classification of Hidden Markov Models

David S. Lalush

Introduction

Hidden Markov models (HMMs) [1-3] have been shown to be quite effective in classifying protein families and recognizing protein motifs from amino acid sequences. These models consist of a set of states, each with a probability of generating a particular residue, and a set of transition probabilities for moving from one state to the next. For protein sequences, there is usually a single state for each residue position in the model, and the transition possibilities are restricted to three: transition to the next residue position, insertion, or deletion of the next position. Thus, HMMs are constrained to represent only correlations that are local in the amino acid sequence. That is, the amino acid observed at a particular position only depends directly on the position immediately preceding it.

Further, the HMM is assumed to be causal in that the dependencies are only with preceding positions and not the following positions. One could turn the sequence around and model it as anticausal, but two-way correlations cannot be modeled with the HMM structure that is usually employed.

These observations lead to questions about whether more distant correlations contain important information that might improve classification performance. From observing protein structures, it is clear that most active sites are composed of residues from different parts of the protein sequence, so they are not as local as the HMM model would imply. Thus, if we had a way to model more distant correlations, we might be able to improve the specificity of motif recognition.

Once we determine that we want to try to model more distant correlations, we encounter two related problems. First, we have to consider how we will model the correlations. Given an unknown sequence aligned to a known model, one straightforward approach would dictate that we develop a joint probability model for each *pair* of residue positions. Such a model would be multinomial, with 400 (20 amino acids x 20 amino acids) possible outcomes. Unfortunately, developing such a model for anything but the smallest protein motifs quickly becomes a large problem. For a 200-residue model, we would have 200x200 = 40000 pairs to evaluate, with 400 possible outcomes for each. Further, as we shall see in a later example, using all of the possible pairs may not give optimal detection.

Given that we do not use all possible pairs of positions, we have to choose a subset of the pairs to use in our evaluation, and therein lies the second problem. We will need a method for determining *which* pairs of positions to use. Fortunately, information theory provides us with a useful tool in this regard. The measurement of *mutual information* [4] between residue positions can help determine which pairs are most informative. This method has been used previously to investigate properties of the basic helix-loop-helix domain [5], but could be used with virtually any protein family with enough known members to compute reliable probabilities.

Mutual information between two random variables x and y measures the degree to which knowledge of x informs about y, and vice versa. If x and y are dependent random variables, then their mutual information is high. If they are independent, then their mutual information is zero. Those pairs with high mutual information may be good candidates for use in motif detection because they represent locations where certain amino acid pairs tend to occur together.

We propose to use the most informative pairs of residue positions, or *cliques*, to enhance the detectability of hidden Markov models. In our approach, an unknown protein is aligned to an

HMM, and then a score for each clique is computed from the multinomial probability derived from the alignment of the known class members to the HMM. The clique scores are summed to generate a composite clique score. We then compute the optimal linear combination of the HMM score and the clique score to give a single metric for the query protein. That metric can be used to classify the query as a member of the family or not.

In the sections that follow, we investigate the utility of the proposed method by examining its classification performance on three protein families: globins, EF-hand proteins, and flavodoxins. We will first describe the methods used to compute the informative cliques from an HMM alignment. That will be followed by the experimental methods used to determine the optimal linear discriminants and the optimal clique size for each protein type. Next we will describe the experiments used to apply the linear discriminant to the complete Swiss-Prot database [6, 7]. We will then follow with the results of these experiments and some discussion of the results.

Methods

Extracting Members of the Protein Families

For each of the three protein families, globin, EF-hand, and flavodoxin, we performed text searches using the Sequence Retrieval System of the Swiss-Prot database [6, 7], release 39.23. The searches were as follows: "globin" returned 827 entries, "ef-hand" or "ef hand" returned 673 entries, and "flavodoxin" returned 107 entries. These entries were downloaded as FASTA-format files. While there may be problems with the text search, i.e., there may be some members of the families missed, we expect that the HMMs derived from the entries retrieved to be representative of the family. This approach has been used previously [8].

For each family, our method requires a representative HMM. For the globin family, we selected 500 sequences from the set of globins, performed a multiple sequence alignment using ClustalW [9, 10], and then input that result into the program *hmmbuild* from the HMMER package [3], obtained from Washington University. For the EF-hand family, we obtained a multiple sequence alignment for the EF-hand domain from the PFAM database [11], and used that as input to the *hmmbuild* program. For the flavodoxin family, we used each of the above methods: one HMM derived from a ClustalW multiple sequence alignment, and one from a PFAM alignment. The two HMMs derived for the flavodoxins were quite different, and we analyzed each of them separately. The properties of the HMMs for each family are summarized in Table 1.

Table 1: Properties of four protein family models used

Name	MSA obtained from:	Number of members	HMM model length
Globin	ClustalW	827	187
EF-hand	PFAM	673	29
Flavodoxin-1	ClustalW	107	646
Flavodoxin-2	PFAM	107	173

After deriving the HMMs, the family members were aligned to their respective HMMs using the program *hmmalign* from the HMMER package. For each family, this generated a multiple sequence alignment aligned to the positions of the HMM model. These multiple sequence alignments were then used to compute the mutual information at each pair of model positions.

Deriving Mutual Information from an HMM Multiple Sequence Alignment

Mutual information *M(i,j)* of two model positions *i* and *j* is computed as follows:

$$M(i,j) = \sum_{d}\sum_{c} \mathrm{P}\left(A_i = c, A_j = d\right) \log_2 \left[\frac{\mathrm{P}\left(A_i = c, A_j = d\right)}{\mathrm{P}\left(A_i = c\right)\mathrm{P}\left(A_j = d\right)} \right] \quad (1)$$

where the summation variables *c* and *d* take on the value of each of the twenty amino acids, P() represents the probability of the event in parentheses, and A_i represents the amino acid at position *i*. Therefore, the expression P(*Ai = c, Aj = d*) represents the joint probability of the amino acid at position *i* being *c* and the amino acid at position *j* being *d*. Because the logarithm is base two, the units of mutual information are *bits*.

For this study, we developed a program in the Python language to compute mutual information from a HMM multiple sequence alignment given by *hmmalign*. In our case, the individual position probabilities were computed from the HMM multiple sequence alignment by taking each position and determining the frequency with which each of the twenty amino acids occurs, as in deriving a position-specific scoring matrix [12, 13]. Only sequences with no gap at the given position were used in the computation.

To compute the joint probabilities, a similar approach was used for the 400 possible amino acid pairs, determining the frequency of each amino acid pair for a given clique. If a gap occurred at either of the two clique positions, that sequence was ignored in the frequency calculation. Mutual information was only computed for positions of the multiple sequence alignment that were aligned to the HMM model, i.e., insertion states were ignored. Also, the mutual information was set to zero for model positions that had more than 50% of the sequences with a gap symbol. This threshold was arbitrary, and its effects should be studied in the future. Finally, single position cliques, i.e., the mutual information of a position with itself, were not used.

Our program will output in a file the *K* most informative cliques from the mutual information computation, and also outputs an NxN greyscale image representing the mutual information at each clique, where N is the length of the HMM. For each of the *K* most informative cliques, the program stores the model positions and the 20x20 joint probability matrix for that pair of positions. The clique file is read by the clique scoring program, described later.

Finding the Optimal Linear Discriminant

To find the optimal linear discriminant for each family which provides the (hopefully) best linear combination of the HMM score and the clique score, we performed a classification test with a small random test set of 500 protein sequences taken from the Swiss-Prot database, version 39. In each case, the random set was checked and revised to eliminate members of the family. For the family members (the true positives) and the random set (the true negatives), the program *hmmpfam* from the HMMER package was run to obtain scores for each protein sequence's alignment with the family HMM. The E-score threshold was set to 10^5 to obtain a score for each protein, even the most poorly aligned ones.

We developed a second Python program to take the output of *hmmpfam* and compute a clique score for each sequence, based on its alignment with the HMM and the set of probabilities in the clique file. The clique score was computed using a log-odds based formula:

$$S_{clique}(A) = 100 + \frac{1}{K} \sum_{cliques(i,j)} \log_{10} \left[\frac{\mathrm{P}\left(A_i, A_j\right)}{1 - \mathrm{P}\left(A_i, A_j\right)} \right] \tag{2}$$

where A is a protein sequence with amino acid A_i at position i in the HMM alignment, $\mathrm{P}(A_i, A_j)$ is the joint probability of the two observed amino acids according to the clique description, and K is the number of cliques. If the argument of the logarithm is less than 1.e-100, the logarithm term is thresholded to –100 for that clique. Thus, the arbitrary addition of 100 to the score makes the clique score tend to zero for poor alignment with the set of cliques. A large positive number would tend to indicate a good alignment with the cliques. Equation (2) contains a number of arbitrary decisions that could affect results, especially the use of log-odds scoring versus other methods. These are issues that should be studied in the future.

For each family, we evaluated the set of true positives and the set of true negatives against both the corresponding HMM and a set of cliques. For each sequence we obtained a pair of scores ($S_{hmm}(A)$, $S_{clique}(A)$) which was treated as a feature vector. We computed the Fisher linear discriminant for each family and number of cliques using methods described in [4]. This gave the linear combination, w, of the two metrics with optimal separation between the positive and negative classes:

$$S_{Fisher} = wS_{hmm} + (1-w)S_{clique} \tag{3}$$

Optimizing the Number of Cliques

The 500-member random test sets were also used to evaluate the effects of number of cliques, K, on class separability. For a given metric (S_{hmm}, S_{clique}, or S_{Fisher}) and family, we computed the *class separability*, sometimes called *discriminability* [4], as follows:

$$d = \frac{m_{positives} - m_{negatives}}{\sqrt{s^2_{positives} + s^2_{negatives}}} \tag{4}$$

where $m_{positives}$ represents the mean value of the metric among all true positive sequences, $s_{positives}$ represents the standard deviation of the metric among all true positives, and similar symbols apply for the true negatives. This is a kind of signal-to-noise ratio representing how far apart the two classes are relative to their distributions. It relates directly to detectability and the receiver-operating characteristic (ROC) for normally-distributed populations. In our case, none of the metrics appears to normally-distributed, so it is not a perfect measure. It is, however, quite simple to compute and is closely related to the optimized separability determined by the Fisher discriminant.

We evaluated each family for a variety of numbers of cliques, and computed the class separability for each. This allowed us to examine the effects of K, and choose an appropriate value of K to be used for scanning the complete database.

Scanning the Swiss-Prot Database

To determine the ROC curve for each case, we extracted from the Swiss-Prot database sets of proteins not included in our protein families. Version 39 of the database includes 86593 proteins, and these family-excluded datasets comprised 85808 sequences for “not_globins”, 86029 sequences for “not_ef_hand”, and 86496 for “not_flavodoxin.” The “not” and family datasets do not generally sum to 86593 because the family datasets were derived from the more recent web-accessible version of the Swiss-Prot database (version 39.23) which contains more

sequences than the downloaded Version 39. No sequences were common to both the family and the "not family" sets.

For each family, a particular value of *K* was chosen based on separability determined in the previous step and on processing time. As increasing K increases the processing time for each sequence, if increasing *K* resulted in only a small improvement in *d*, the smaller value was chosen. We used *K*=1000 for the globins, *K* = 400 for the EF-hand, and *K* = 500 for both flavodoxin sets.

We evaluated all members of the family and "not family" sets using first *hmmpfam* and then our clique scoring program. We then computed the Fisher metric for each sequence using the discriminant determined from the small test set. For each metric, the entire set of proteins were ordered according to score, and true positive fractions and false positive fractions were computed by taking intervals of the ordered set, and computing the number of true positives and false positives at each interval. From these data, we were able to plot ROC curves for each case.

Results

Visualization of Mutual Information

Figure 1 presents images that provide a useful visualization of mutual information for each protein family. The images are reminiscent of dot plots, with each pixel representing one clique. Consider the model positions as lying across the top of the image and down the left side of the image. Each pixel represents the pair of positions that intersect at its location. The greyscale value of the pixel is related to the mutual information for that clique. To make the images more useful, we show only the *K* most informative positions with all others set to black. The diagonals are also set to black because single-position cliques are not used in our model.

One important observation found in all of the images is that many of the most informative positions are far from the diagonal. This indicates that these correlations are rather distant in terms of the sequence, since local correlations would be found close to the diagonal. Further, we note that many of the most informative positions fall in certain columns or rows of the image. Thus, certain positions tend to form more informative cliques with a number of other positions. Interpreting the significance of such positions is difficult. They cannot be absolutely conserved, because the mutual information of any clique containing a perfectly conserved position is zero. (This occurs since the two positions would meet the definition of independence in such a case.) Such positions would likely have several co-occurring pairs of amino acids. Future studies should examine the structural and functional significance of such sites.

We expected to find that certain cliques would stand out with relatively high mutual information, but this was not the case for any of our models. Many cliques were close to the maximum mutual information value, so we could not extract particularly sensitive locations. Thus, we expect that it will be necessary to use a relatively large number of cliques for recognizing proteins.

The image for the EF-hand family is particularly interesting because it illustrates properties of the EF-hand structural motif. This motif consists of 9-12 residues of alpha helix followed by 9-12 residues of random coil and 9-12 residues of another alpha helix [14]. We see that the mutual information image for the EF-hand divides the sequence into three roughly equal sections. The most informative cliques tend to involve the alpha helices with the coil section in the middle tending to have low mutual information. The implication is that the alpha helices

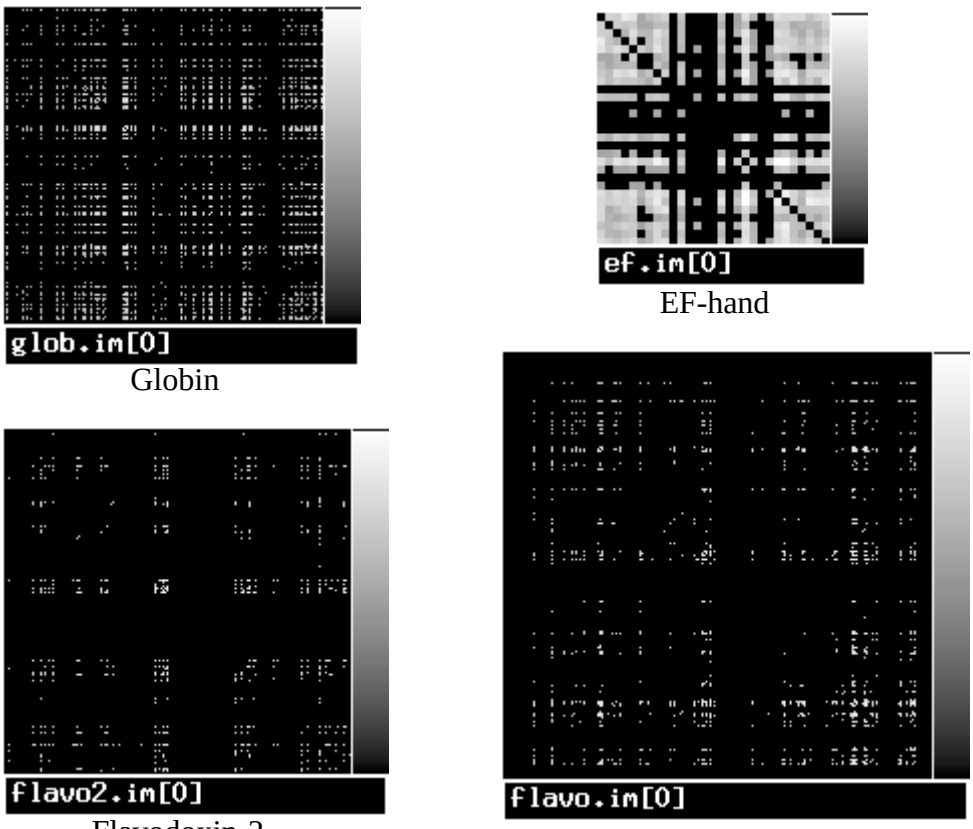

Figure 1: Mutual information maps for the four protein models. The maps have been thresholded for the K highest cliques, where K = 2000 for the globin model, 400 for EF-hand, and 500 for both flavodoxin models. Note that flavodoxin-1 only maps model positions 400-646 as these contained all of the highest values of mutual information.

tend to have dependent, co-occurring amino acids while the amino acids in the coil section tend to vary independently.

Effects of Number of Cliques

Figures 2 through 5 show the effect of number of cliques on class separability for each of the four models studied. In general, we find that the clique scoring alone does not provide better separability than the HMM, although the flavodoxin-1 model is an exception. Further, we find that, while there is usually a *K* value at which the separability is optimized, the optimum is usually not very distinct and the *K* value can be reduced for computational efficiency without sacrificing separability much.

The separability values can be used as a predictor of the comparative performance of the methods in detection of the protein family. Thus, we would expect the Fisher metric to significantly outperform the HMM for the EF-hand case, but probably not for the globin case. However, we must be careful with such predictions for two reasons. One, the detectability metric is only an accurate representation when the distributions are normal, and, from observation of the histograms, ours are not. Two, we have not computed the standard errors on the detectability metric, so we do not know if the differences are statistically significant. It

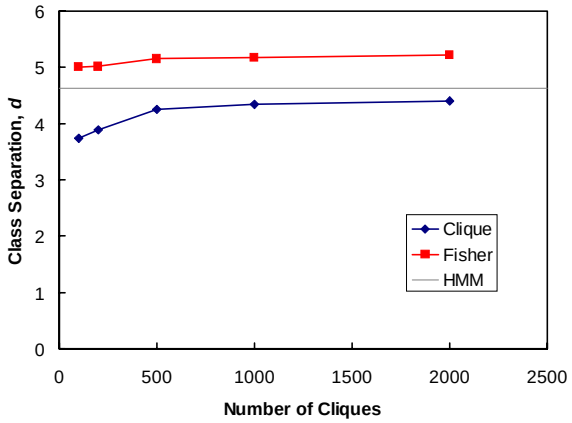

Figure 2: Plot of class separation versus number of cliques for the globins model.

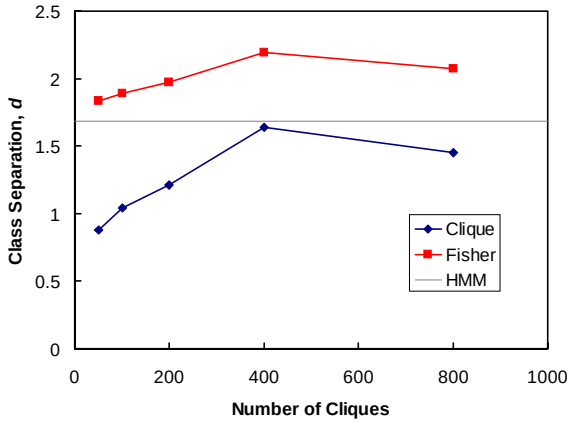

Figure 3: Plot of class separation versus number of cliques for the EF-hand model.

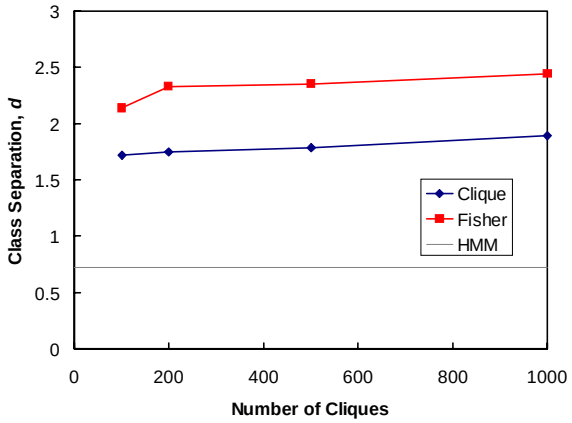

Figure 4: Plot of class separation versus number of cliques for the flavodoxin-1 model.

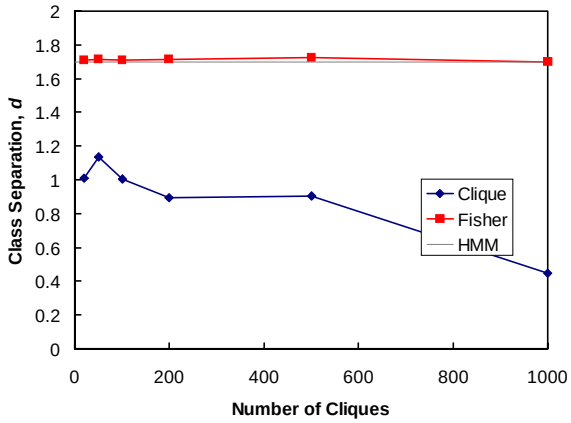

Figure 5: Plot of class separation versus number of cliques for the flavodoxin-2 model.

would be possible to do that computation, or to fit an ROC curve to the test-set data to estimate significant differences in performance. We did so using the program *Rockit* from University of Chicago [15], and found significant differences only in the flavodoxin-1 model between clique and HMM and between Fisher and HMM. Thus, we expect to see these differences in performance in database scanning.

The individual results for the EF-hand family show an interesting peak at $K = 400$ and a reduction in performance at $K = 800$. Because the model is only 29 residues long, there are only $29^2 = 841$ cliques available. Thus, performance is degraded by including nearly all of the cliques in this case. So, even if we chose to pay the computational costs of using all of the cliques, it may not be the best choice to do so. For the other, longer models, we did not test for K being close to the total number of cliques available, so we cannot conclude that this effect necessarily continues to hold for other families.

Database Scanning Performance

Figures 6 through 9 present the ROC curves for the four models. For each case, we present a close-up view of the curves for false positive fractions up to .1, and a second view for the full range of false positive fraction up to 1.0.

Figure 6 gives the scanning results for the globin model. We find that the Fisher metric consistently has a lower ROC curve than the HMM model. An exception to this is for the first 500 proteins scanned (This is difficult to see in Figure 6.), Fisher gets about three more hits than the HMM. The results are, however, pretty conclusive that the Fisher metric does not improve performance of the globin HMM. One possible reason is that the distributions of the globin HMM and clique scores are far from normal, and so the Fisher method cannot find the true optimal discriminant. A second possibility is more rooted in the properties of the globin family: the globin alignment may contain a relatively large number of well-conserved amino acids. These are modeled effectively by the HMM, but not by the clique score, which tends to favor consistently-occurring pairs. Thus, the globin family may have properties that simply do not lend themselves to detection by the clique score.

Figure 7 shows results for the EF-hand model. In this case, the Fisher model outperforms the HMM for low levels of FPF (up to the highest-scoring 8000 proteins), but then the curves cross and HMM performs as well or better thereafter. This is, of course, our smallest model and is representative of the task of finding relatively small protein domains. Again, the assumptions of normality implicit in the use of the Fisher metric are probably violated, so we may not have truly optimal performance for the linear discriminant. However, the Fisher model does give improved performance in finding the first 95% or so of EF-hand proteins. Thus, clique scoring data does add useful information for the task of finding the most homologous proteins in this case. If the task is to find the last few family members, then the Fisher metric does not perform as well as the HMM.

Figure 8 gives results for the flavodoxin-1 model, derived from ClustalW multiple sequence alignment. This was a particularly long model, and it gave some unexpected results. Namely, we found that both the clique scoring method and Fisher far outperformed the HMM. Note that this was predicted by our small test set analysis in that these were the only cases where a

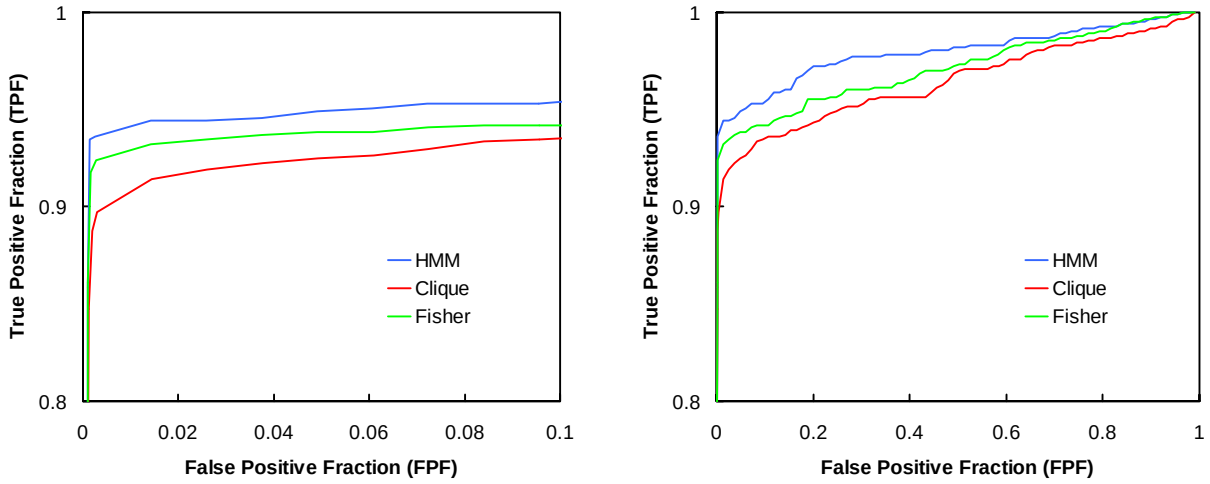

Figure 6: ROC curves from scanning the Swiss-Prot database for the globins model. The left chart presents a close-up view of FPF up to .1; the right presents the full curves.

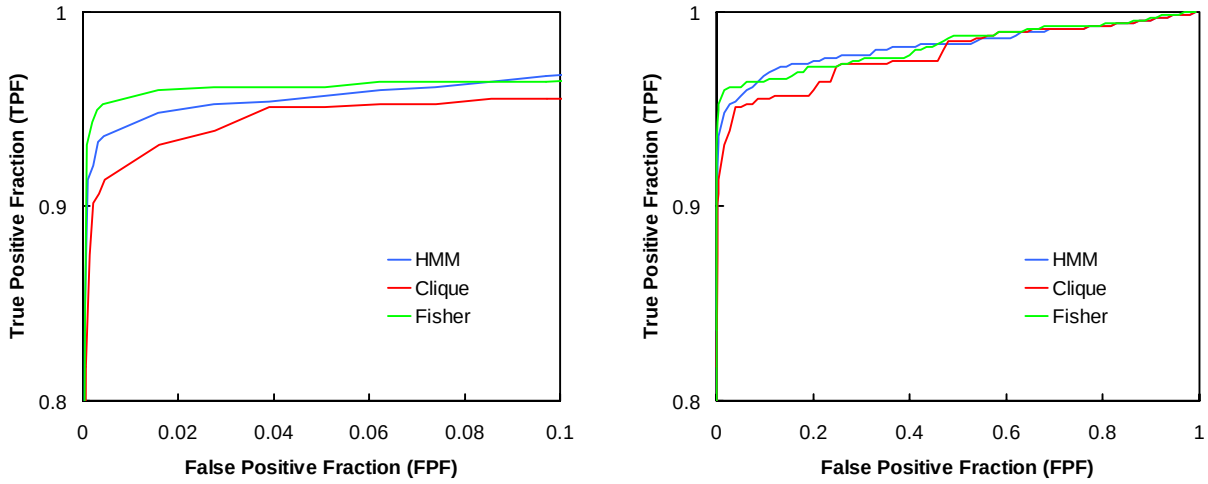

Figure 7: ROC curves from scanning the Swiss-Prot database for the EF-hand model. The left chart presents a close-up view of FPF up to .1; the right presents the full curves.

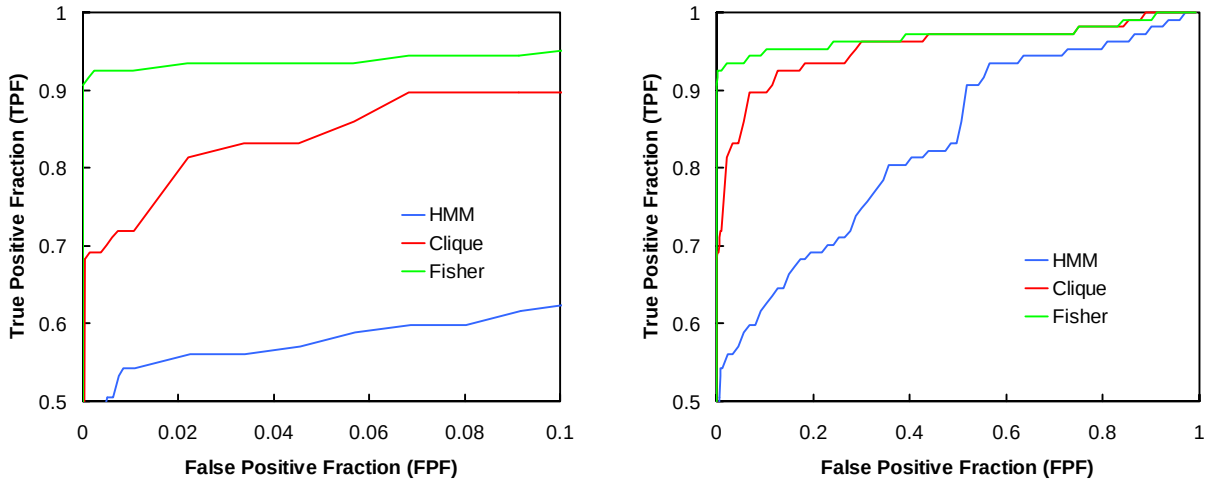

Figure 8: ROC curves from scanning the Swiss-Prot database for the flavodoxin-1 model. The left chart presents a close-up view of FPF up to .1; the right presents the full curves.

significant difference in areas under the ROC curves were found. In looking at the HMM alignments, it became clear that this HMM model was particularly poor, with many members of the flavodoxin family generating poor scores. One would expect that a poor HMM model would lead to poor performance of the clique score and consequently the Fisher metric. But, in fact, the Fisher metric performs very well. The results become even more puzzling when we look at the second flavodoxin model.

Figure 9 gives the results for the flavodoxin-2 model, derived from the PFAM alignment. In this case, the more compact HMM performed better than the flavodoxin-1 model. However, the clique scoring method performed *worse* than that from the flavodoxin-1 model. The Fisher metric does still consistently outperform the HMM, though not as significantly as with flavodoxin-1. It is encouraging that, for the flavodoxins, the Fisher metric provides improvement in classification performance at all levels of FPF, for two different HMMs.

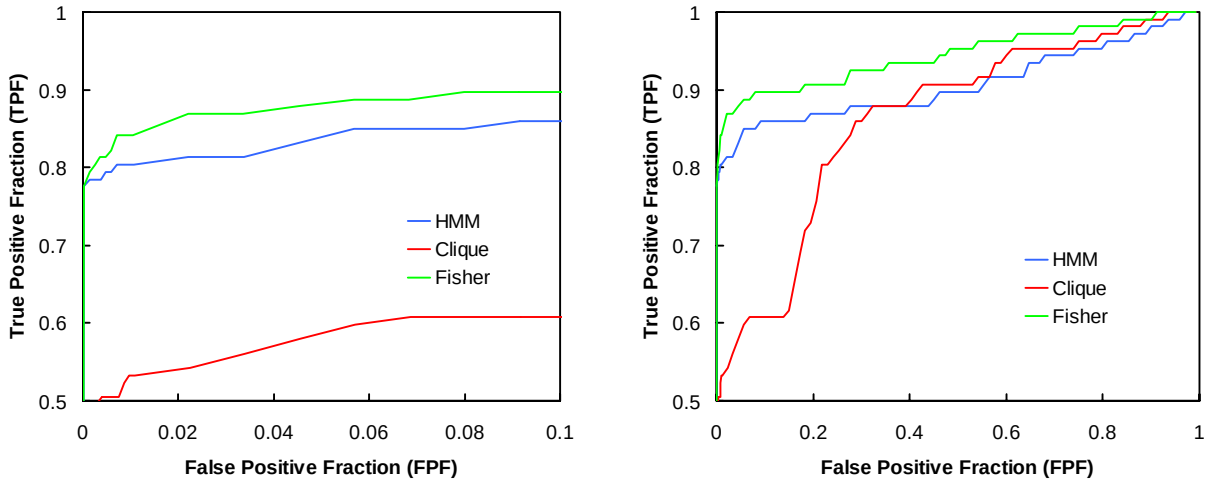

Figure 9: ROC curves from scanning the Swiss-Prot database for the flavodoxin-2 model. The left chart presents a close-up view of FPF up to .1; the right presents the full curves.

We note that the Fisher model for flavodoxin-1 shows better performance than the Fisher model for flavodoxin-2, even though the performance comparison of the HMMs on which they are based is opposite. This leaves quite a mystery that further analysis of the properties of these multiple sequence alignments, HMMs, and mutual information maps should try to explain. It may give us an indication of what properties of protein families and HMMs are best suited to use of clique scoring.

Discussion

Among the three protein families studied, we have mixed results for the use of clique scoring and a Fisher linear discriminant to enhance the selectivity of protein family recognition. For the globins, the clique scoring did not improve the performance of the HMM in any way. For the EF-hand proteins, clique scoring improved on the HMM performance in finding the first 95% of members of the protein family, but was not as good thereafter. Finally, with the flavodoxins, clique scoring consistently improved upon the ability of the HMMs to recognize family members. Based on these results, we can conclude that clique scoring may enhance classification performance, but the results will depend on which protein family we are trying to detect. A useful future direction for this work would be to examine other protein families in this way, and to attempt to find some properties of protein families that predict when clique scoring will improve or degrade performance.

This is merely a first attempt at such an application of distant correlations among amino acid positions. In [5], the authors used entropy analysis of single positions along with mutual information analysis to study the properties of basic helix-loop-helix (bHLH) proteins. However, they used this information to develop a regular expression for database scanning. A problem with the use of regular expressions is that they do not capture the probabilistic nature of the correlations, and thus have no way of assessing the significance of matches. Our approach improves on this idea by applying methods similar to position-specific scoring matrices. A position-specific scoring matrix contains a multinomial probability distribution for each residue position. In our case, we have simply added another dimension to this idea to evaluate pairs of positions together. The clique scores are directly related to the observed probabilities of the co-

occurring amino acids. With further work on the statistics of clique scoring, it should be possible to provide significance estimates of both the clique scores themselves and the Fisher scores.

We have made a number of arbitrary, though hopefully logical, choices in our implementation of clique scoring. These include the functional form for computing the clique scores (Eq. (2)), the methods for determining the linear discriminant, and even the use of HMMs as the basis for multiple sequence alignment and scoring. There are other options for all of these, and other choices would likely change the performance of the method. Further research should examine and evaluate other approaches. For example, clique scoring would naturally fit in with a position-specific scoring matrix, so that it would be possible to align an unknown protein directly with a family described by both single-position *and* clique scores. Because of their two-way, non-causal nature, aligning to clique scores would require the use of a stochastic algorithm that randomly generates and evaluates different alignments.

Further, it would be possible to define cliques containing more than two positions, because these would simply be multinomial distributions also. Mutual information could not be used to evaluate such cliques, so other measures, perhaps joint entropy, would have to be used to find good candidates. Also, the number of possible outcomes increases exponentially with clique size, so larger-sized cliques would have to be chosen very carefully to find those that are most helpful in identifying protein motifs.

In forming our probabilities related to clique outcomes, we used only the observed frequencies directly from the multiple alignments. We did not consider issues related to species or protein bias in the families. These arise from the fact that only certain species have been studied extensively, and so the protein databases contain large numbers of proteins from these and fewer from other species. Similarly, certain homologous proteins may be more present in the database and therefore may bias the results of the HMM and the clique probabilities. To address this, one could consider weighting the sequences differently [8] and/or including prior information in the form of pseudocounts in the clique frequency matrices [12]. These methods could help remove biases and improve the performance of clique scoring.

Another option in the implementation of clique scoring is to alter the definition of the discrete events represented by the amino acids. In [5], the authors proposed classifying the amino acids into eight types (acidic, basic, aromatic, etc.) thus reducing the number of outcomes at each site from twenty to eight. Mutual information is then computed in the same way, with the transformed set of outcomes. This would tend to give a more biochemically oriented view, and would allow substitutions of chemically-similar amino acids. Such a method may be more forgiving, especially if the family in question has relatively few members. In the future, the method we propose here should be compared to the use of amino acid classes to determine if performance can be further improved.

Computational load is an important consideration in any database scanning method. The clique evaluation generally took about half again as much time as the original HMM alignment, so it can be a significant computation. It is likely that the clique scoring could be improved with optimization techniques, as we did not seek to maximize efficiency here. But, the computation time will increase linearly with K, so it will always be important to choose K to be as small as possible without sacrificing too much performance.

Conclusions

We have developed methods for enhancing protein family and motif recognition by evaluating cliques, or highly informative pairs of protein positions. Our method for clique

identification, using mutual information, and clique scoring, akin to a position-specific scoring matrix, was combined with scores from hidden Markov models via a linear discriminant. Optimization of the number of cliques used in clique scoring may be necessary in some cases to obtain best performance. In database-scanning tests against three protein families, our method was found to improve on the use of HMM alone in some cases and not in others. We conclude that the use of informative cliques can improve the performance of hidden Markov models, but that results depend on a number of factors, and especially on the properties of the protein family itself.

References

[1] M. Brown, R. Hughey, A. Krogh, I. S. Mian, K. Sjolander, and D. Haussler, "Using Dirichlet mixture priors to derive hidden Markov models for protein families," *Proc ISMB*, vol. 1, pp. 47-55, 1993.

[2] S. R. Eddy, "Hidden Markov models," *Current Opinion in Structural Biology*, vol. 6, pp. 361-365, 1996.

[3] S. R. Eddy, "Profile hidden Markov models," *Bioinformatics*, vol. 14, pp. 755-763, 1998.

[4] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: John Wiley and Sons, 2001.

[5] W. R. Atchley, W. Terhalle, and A. Dress, "Positional dependence, cliques, and predictive motifs in the bHLH protein domain," *Journal of Molecular Evolution*, vol. 48, pp. 501-516, 1999.

[6] A. Bairoch and R. Apweiler, "The SWISS-PROT protein sequence database: its relevance to human molecular medical research," *Journal of Molecular Medicine*, vol. 75, pp. 312-316, 1997.

[7] A. Bairoch and R. Apweiler, "The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999," *Nucleic Acids Research*, vol. 27, pp. 49-54, 1999.

[8] R. Karchin and R. Hughey, "Weighting hidden Markov models for maximum discrimination," *Bioinformatics*, vol. 14, pp. 772-782, 1998.

[9] D. G. Higgins and P. M. Sharp, "CLUSTAL: a package for performing multiple sequence alignment on a microcomputer," *Gene*, vol. 73, pp. 237-244, 1988.

[10] J. D. Thompson, D. G. Higgins, and T. J. Gibson, "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice," *Nucleic Acids Research*, vol. 22, pp. 4673-4680, 1994.

[11] A. Bateman, E. Birney, R. Durbin, S. R. Eddy, K. L. Howe, and E. L. Sonnhammer, "The Pfam protein families database," *Nucleic Acids Research*, vol. 28, pp. 263-6, 2000.

[12] J. G. Henikoff and S. Henikoff, "Using substitution probabilities to improve position-specific scoring matrices," *Computer Applications in Bioscience*, vol. 12, pp. 135-143, 1996.

[13] S. Henikoff, "Scores for sequence searches and alignments," *Current Opinion in Structural Biology*, vol. 6, pp. 353-360, 1996.

[14] H. Kawasaki and R. H. Kretsinger, "Calcium-binding proteins 1: EF-hands," *Protein Profile*, vol. 2, pp. 305-490, 1995.

[15] C. E. Metz, "Some practical issues of experimental design and data analysis in radiological ROC studies," *Investigative Radiology*, vol. 24, pp. 234-245, 1989.