PEDS Advance Access published July 2, 2009

Protein Engineering, Design & Selection pp. 1–7, 2009
doi:10.1093/protein/gzp035

Discovering rules for protein–ligand specificity using support vector inductive logic programming

Lawrence A. Kelley[1,3], Paul J. Shrimpton[1], Stephen H. Muggleton[2] and Michael J.E. Sternberg[1]

[1]Structural Bioinformatics Group, Division of Molecular Biosciences and [2]Computational Bioinformatics Laboratory, Department of Computing, Imperial College London, London, UK

[3]To whom correspondence should be addressed.
E-mail: l.a.kelley@imperial.ac.uk

Structural genomics initiatives are rapidly generating vast numbers of protein structures. Comparative modelling is also capable of producing accurate structural models for many protein sequences. However, for many of the known structures, functions are not yet determined, and in many modelling tasks, an accurate structural model does not necessarily tell us about function. Thus, there is a pressing need for high-throughput methods for determining function from structure. The spatial arrangement of key amino acids in a folded protein, on the surface or buried in clefts, is often the determinants of its biological function. A central aim of molecular biology is to understand the relationship between such substructures or surfaces and biological function, leading both to function prediction and to function design. We present a new general method for discovering the features of binding pockets that confer specificity for particular ligands. Using a recently developed machine-learning technique which couples the rule-discovery approach of inductive logic programming with the statistical learning power of support vector machines, we are able to discriminate, with high precision (90%) and recall (86%) between pockets that bind FAD and those that bind NAD on a large benchmark set given only the geometry and composition of the backbone of the binding pocket without the use of docking. In addition, we learn rules governing this specificity which can feed into protein functional design protocols. An analysis of the rules found suggests that key features of the binding pocket may be tied to conformational freedom in the ligand. The representation is sufficiently general to be applicable to any discriminatory binding problem. All programs and data sets are freely available to non-commercial users at http://www.sbg.bio.ic.ac.uk/svilp_ligand/.

Keywords: functional residues/SVILP/machine learning/protein structure/function prediction

Introduction

With the sequencing of the human genome and many other organisms, rapid determination of gene and protein function is becoming increasingly important. Structural genomics initiatives are helping elucidate the function of these gene products by developing high-throughput methods for determining structures for all unique protein folds. These new structure targets are often specifically selected not to have sequence similarity to existing proteins (Burley *et al*., 1999; Skolnick *et al*., 2000; Brenner, 2001; Baker and Sali, 2001). With the anticipated explosion of available structures, it is imperative to develop computational methods for high-throughput function prediction on protein structures.

Sequence motifs

There has been extensive work in identifying conserved residues in protein sequences with similar function. Amino acid sequence patterns that represent these conserved residue positions can be created from multiple alignments of sequences with similar function. These patterns, or sequence motifs, can be used to assign function to sequences that contain the pattern. Numerous sequence motif databases have been established with different methods for creating sequence motifs. Some of the databases are manually curated by experts, whereas others are automatically derived (Henikoff *et al*., 1999; Huang and Brutlag, 2001).

Although sequence motifs can provide insight into protein function, when novel proteins do not share significant sequence similarity with proteins of known function, sequence information alone is insufficient for functional annotation. Proteins that do not have high sequence similarity may still have similar function because of conservation of physicochemical properties at the structural level (for a review, see Sadowski and Jones, 2009).

Function from structure

Proteins of known structure, but of unknown function, are typically compared with databases of other structures to discover functional relationships. Methods such as DALI (Holm and Sander, 1995) or VAST (Gibrat *et al*., 1996) perform structural alignments to a database of known structures and in order to find proteins with a similar fold. Such similarities can identify ancient evolutionary relationships that are not always apparent when only sequences are known, but that are often associated with a similarity in function.

However, search methods based on structural alignment do not always provide functional clues. This is clear if a protein adopts a new fold, but problems can also arise when proteins adopt common folds that perform many different functions, such as a TIM-barrel, ferredoxin or immunoglobulin-like structures (Orengo *et al*., 1994). Here functional inferences are difficult to make, as equally close structural alignments can be generated between functionally similar and dissimilar proteins.

An alternative strategy is to obtain functional clues by detecting local structural patterns associated with a particular function. Residues within these patterns are not necessarily adjacent in the protein sequence and can occur in any order. A classic example is the trypsin-like catalytic triad, which nature has reinvented more than 10 times (Dodson and Wlodawer, 1998). These functionally important similarities cannot be detected by sequence comparison or structural alignments and require methods that are independent of sequence or fold similarity.

Structural motifs

Analogous to sequence motifs, structural motifs provide a description of conserved properties in the three-dimensional structure of proteins sharing a molecular function. Investigators have devised different techniques to construct and define structural motifs; each technique emphasises different conserved properties.

Wallace *et al*. (1996, 1997) have developed a system, PROCAT, for identifying catalytic sites by geometric orientation of residues with known functional importance. By using previous knowledge of the critical residues involved in the catalytic activity, a structural motif representing the conserved relative positions of those residues is constructed. This motif can be used to scan a new protein structure for occurrence of the catalytic site using a geometric hashing algorithm.

Fetrow and Skolnick have developed fuzzy functional forms for representing distances between key residues. The critical residues involved in a functional site are identified by careful examination of the literature. Examples of known structures containing these residues are used to find mean distance and variance between the residues. The structural motif representing the conserved distance and variance of the residues is used to identify functional sites on protein structures (Fetrow and Skolnick, 1998; Fetrow *et al*., 1998). Laskowski *et al*. (2005) use a combination of handcrafted and automatically generated fragments of protein structures as a library of motifs. Together with carefully calibrated statistics to overcome spurious matches, this technique is powerful, although limited by the requirement of homology between proteins with a common binding site. Zhao *et al*. (2001) have used the consensus values of a grid-based energy function together with docking to predict adenylate binding sites. The PINTS system (Stark and Russell, 2003) matches structural subsets of a protein (amino acids close in space) to either (i) a library of pre-compiled functionally interesting subsets from known structures or (ii) an individual protein structure. Conversely, one can match a library of pre-compiled subsets to a protein of interest.

A frequent problem in protein function prediction lies in the difficulty in distinguishing between proteins with similar functions but different ligand specificities. Subtle differences in a binding pocket can shift the specificity of that pocket from one ligand to another. For this reason, we have chosen two common ligands for this study, FAD and NAD. The ubiquity of these ligands across a wide range of protein structures and functions provides us with a large and diverse data set of experimentally derived structures with the ligands bound.

A complete definition of the geometry and composition of a binding pocket can be gained by tabulating the set of all pairwise inter-residue distances across the pocket. However, we would like to discover if there are rules or key features of certain classes of binding pocket which determine the ligand specificity of that pocket. One would expect there to be some common key features that are shared between pockets that bind one ligand over another. It is our aim in this paper to describe the development of a method which automatically determines such key features and which demonstrates high accuracy (80+%) in classification of novel examples.

Although pairwise distance terms contain much of the information available about the binding site, we expect higher order relationships to more succinctly capture the ligand specificity, as these relationships better reflect the shape and biophysical properties of the ligand. For example, a triplet of amino acids in the pocket forms a triangle of biophysical properties with the residues as the vertices of the triangle. Unfortunately, when considering higher order relationships such as triplets of properties, the search space of potentially interesting/useful triplets quickly grows unmanageable. A given triplet such as [alanine, glutamate, tyrosine] together with their three distances can be expressed in many ways depending upon the biophysical representation: [small, charged, bulky], [small, glutamate, aromatic], [nonpolar, negatively charged, polar/aromatic]. . . , etc. A rough calculation of the number of possible triangles given a conservative binning procedure of distances quickly runs into the millions. When searching for a general principle in a large data set, one would like to be able to examine each of these possibilities, yet the computational burden is often too great.

Relational learning

Relational learning is one powerful approach to solve this problem. In this work, we use inductive logic programming (ILP) (Muggleton and De Raedt, 1994) as a means to learn complex rules about the relations between entities while including the background knowledge of the biophysical properties of the amino acids. In a biological context, ILP has previously been successfully employed for automatic identification of chemical substructures that can be used to describe the toxicity or activity of a compound (King *et al*., 1996; Finn *et al.*, 1998; Sternberg and Muggleton, 2003), in the classification of protein folds (Cootes *et al.*, 2003), in modelling features of metabolic networks (Tamaddoni-Nezhad *et al*., 2006) and in the 'robot scientist' (King *et al*., 2004).

Recently, a new hybrid technique has been developed, known as support vector ILP (SVILP) (Muggleton *et al.*, 2005). This technique lies at the intersection of two areas of machine learning, namely, support vector machines (SVMs) and ILP. It is a novel machine learning approach which combines the dimensionality-independence advantages of SVMs with the expressive power and flexibility of ILP. To date, SVILP has been used on biological data sets to predict bioactivity of small molecules (Cannon *et al*., 2007), quanititative toxicology (Amini *et al*., 2007a) and in the prediction of binding affinities of protein–ligand complexes (Amini *et al*., 2007b).

In this work, we have developed a framework for applying ILP, SVMs and the hybrid SVILP approaches to the problem of predicting ligand specificity. Using only the pairwise distances between residues comprising the pocket and a set of descriptions of the properties of the 20 amino acids, we can use ILP to search for triangles of properties that distinguish one binding pocket from another. These automatically defined rules then form the attributes used as input to an SVM. We find that the performance of the SVM and ILP alone are comparable, although ILP uses only a handful ($\sim$10) of rules compared with the thousands of attributes for the SVM. The use of the hybrid SVILP approach demonstrates a minor increase in recall and comparable precision to either approach alone. However, further analysis of the ILP-derived rules suggests possible links between ligand

flexibility and patches of biophysical properties within the protein binding site. Finally, we develop a novel approach to SVILP (frequency-based SVILP, *f*-SVILP) which permits a radical reduction in the number of attributes required while achieving even higher accuracy of discrimination than any of the other methods.

Methods

Data set generation and cross-validation

Proteins known to bind FAD and NAD were extracted from the MOAD database (Hu *et al.*, 2005) with resolution <2 Å using the 'no redundancy' flag from MOAD. These proteins were grouped according to their first two enzyme classification (EC) numbers. After excluding cases where it was not possible to make unambiguous assignments of EC numbers, this led to a total of 57 FAD-binding structures which could be grouped into 14 unique classes according to the second EC number. For NAD, 40 proteins were extracted which were grouped into 13 classes. A 20-fold leave-one-out cross-validation was performed. In each case, a randomly chosen class based on the first two EC numbers was set aside for testing for FAD-binders and NAD-binders. The remaining classes were used for training.

In this work, we have restricted our analysis to include the flavin ring down to the first phosphate group for FAD, and similarly to include the nicotinamide ring down to the first phosphate group for NAD, i.e. we have excluded the adenylate ring common to both ligands. For each protein, any amino acid with an atom within 5.5 Å of the NAD or FAD moiety in the crystal structure was considered part of the binding pocket. This threshold was set to capture not only directly contacting ligands, but also some sense of the more general chemical environment of the pocket. For a given binding pocket in a given protein, all pairwise distances between the Cβ atoms (Cα for Gly) of the amino acids forming the pocket were tabulated. Thus, any given pocket was represented by the identity and distances between all pairs of amino acids comprising the pocket together with whether that pocket binds NAD or FAD. Thus, side-chain conformations are completely ignored.

Throughout this work, SVMlight (Joachims, 1999) was used with a simple linear kernel and default parameters.

Inductive logic programming

Introduction ILP learns from known examples or observations (i.e. it employs inductive reasoning) (Muggleton and De Raedt, 1994). The observations, the background knowledge and the resultant rules are expressed as first-order logic programs, such as 'FAD binding site example identifier 105 contains a valine residue 5.3 Å from an alanine residue'. CProgol is a state-of-the-art implementation of ILP. CProgol's input consists of positive and negative examples; in this case, FAD-binding pockets and NAD-binding pockets, respectively, together with background knowledge (see below).

The output of CProgol is a set of logic rules which describe the positive and negative examples using the information provided in the background knowledge. In CProgol, the first positive example is randomly selected, and on the basis of the background knowledge, hypotheses are constructed; then, the hypothesis with maximum compression is selected as the result of the search. Compression, C, for each clause is defined as:

$$C = \frac{P[p - (n - l)]}{p}$$

where C, P, p, n and l are compression, total number of positive examples, number of positive examples covered by the clause, number of negative examples covered by the clause and the length of clause (the number of features in each rule), respectively. Compression is a suitable measure for finding those rules which have predictive power, and it avoids overly specific rules (i.e. long clauses). The calculation is continued on the next positive example, but the redundant examples relative to the previously learned rules are removed. One of the advantages of the logic-based method is that it both constructs and selects the hypotheses. The selection is based on the value of compression that is defined automatically for each rule.

At the end of the ILP calculation, a small number of rules (often between 5 and 20 in this work) are produced which cover the positive examples in the training data. These rules can then be used to assess an unseen test set.

Representing binding pockets as background knowledge in ILP In this work, the background knowledge consists of the pairwise distances between amino acids in the pocket, some simple biophysical properties of each amino acid, such as 'Alanine is small and hydrophobic', and the concept of a 'triangle of properties'. Pairwise distances are permitted a rather large flexibility term (± 2.5 Å) to encourage the generation of very general, rather 'soft' rules. Initially, a binding pocket is represented as a series of clauses indicating the type of each amino acid in the pair and their distance apart. The ILP program then searches for triangles of residues, and may substitute a given residue type with a more general biophysical property. Thus, a series of statements such as:

Dist(Val123, val, Glu35, glu, 18.6)
Dist(Val123, val, Arg46, arg, 10.1)
Dist(Arg46, arg, Glu35, glu, 11.7)

represent the pairwise distances and amino acid types comprising the binding pocket. (The first statement can be translated as 'valine 123 is 18.6 Å from Glutamine 35'.) These statements can be automatically recast by the ILP program into a statement about a triangle of properties, e.g. *triangle*(*hydrophobic, polar, polar, 18.6, 10.1, 11.7*) or *triangle*(*hydrophobic, positively_charged, polar, 18.6, 10.1, 11.7*), etc. During the search through hypotheses, certain triangles with certain biophysical representations will generate better or worse compression of the training set.

Support vector inductive logic programming The result of applying ILP is a set of rules (hypotheses) with a range of predictive power. These rules can be considered as attributes for input to an SVM (Vapnik, 1995). Unlike the conventional use of SVMs where an investigator determines what features are likely to be relevant to predictive performance, with SVILP, the features are automatically discovered by the ILP procedure. The principle is to use the truth or falsity of the

rules discovered by ILP as binary attributes for input to a conventional SVM. Thus, given n rules discovered by ILP, an example case can be described by an n-dimensional binary attribute vector, where the attribute is '1' if the rule is true and '0' otherwise. Given the binary nature of the attribute vector, a simple linear kernel has been used in this work together with the SVMlight package (Joachims, 1999).

Simple SVM approach

It is of course important to compare the SVILP approach with a generic SVM approach. A typical approach to handling three-dimensional data is to calculate the frequency with which different distances occur between each possible amino acid pair. Thus, for every protein pocket, a high-dimensional attribute vector was generated. We used a 2.5 Å distance bin given all possible pairs of amino acid types with a ceiling and floor of distance based on the minimum and maximum observed distances across the pockets in the data set. The number of times a particular pair of amino acids was observed at a particular separation was normalised by the total number of distances calculated across the pocket.

Frequency-based ILP

In addition to the conventional SVILP approach, we used a novel approach where the attributes are based on the number of times a given rule is true for a particular example. For a particular binding pocket, a flexible triangle of properties may be present multiple times, with different amino acids forming the vertices of the triangle. This is because the rule may refer to a relatively general biophysical property, such as 'hydrophobic', and the distances defining the triangle are permitted a broad tolerance of ± 2.5 Å. Thus, there may be multiple instantiations of a given rule within a single pocket. These correspond to multiple 'proofs' in ILP. The number of proofs or 'hits' of a rule can then be used instead of the simple binary representation in the attribute vector.

Results

We first compare the classification performance in a 20-fold leave-one-out cross-validation of three methods: ILP, simple SVM and the hybrid, SVILP (Table I).

Pure ILP

The result of learning is a set of rules (in this case, usually between 5 and 15 rules) which cover the training set with high accuracy. For each cross-validation, the rules learnt in training are applied to the corresponding test set. The classification performance is aggregated across the 20 leave-one-out runs and presented in Table I. We can see that ILP performs well on the data, achieving 85% and 83% precision and recall, respectively.

Rules have the general syntax: *FADbinding:-Trip[a,b,c, dist(ab),dist(ac),dist(bc)]*. This indicates that a binding pocket binds FAD, if it contains a triplet of residues a, b and c, where $dist(ab)$ is the distance in Angstroms between a and b. A typical rule, in this case, the one with highest compression in one leave-one-out trial, had the following form: FADbinding: -trip(hydrophobic, hydrophobic, polar, 16.72, 14.89, 9.84).

It is important to remember the flexibility term mentioned in the Methods section. Each of these distances may vary by ± 2.5 Å about the stated value. This single rule was able to correctly classify 29 of the 56 positive cases in this cross-validation with no false positives.

Simple SVM

Using the approach described in the Methods section, each binding pocket was represented as an attribute vector containing the relative frequencies of every possible pair of amino acids in distance bins of 2.5 Å. This resulted in 2100 attributes per vector. Using this approach, the SVM performs comparably with ILP, achieving 81% and 84% precision and recall, respectively.

Support vector inductive logic programming

Those rules generated by ILP with high compression are candidates for attributes in the vector presented to an SVM. We investigated how the number of rules used to form the attribute vector effect performance on an independent 5-fold cross-validation and this can be seen in Fig. 1. As large numbers of rules are included in the attribute vector, performance declines slowly. This is to be expected as the progressively less powerful (lower compression) rules provide progressively less discriminatory signal and thus add less, or can even pollute, the SVM. On the basis of these results, we

Table I. Benchmark results of the various learning approaches on the 20-fold leave-one-out data set for FAD/NAD binding discrimination

	Number of attributes/rules	Precision	Recall
Pure ILP	10 (on average)	85% (100, 18, 67, 21)	83%
SVM	2100	81% (102, 24, 61, 19)	84%
SVILP	300	83% (105, 22, 63, 16)	87%
Frequency-based SVILP	20	90% (104, 12, 73, 17)	86%

Numbers in parentheses represent true positives, false positives, true negatives and false negatives, respectively. Using a one-tailed sign test, frequency-based SVILP is statistically superior to all other methods at $P < 0.05$.

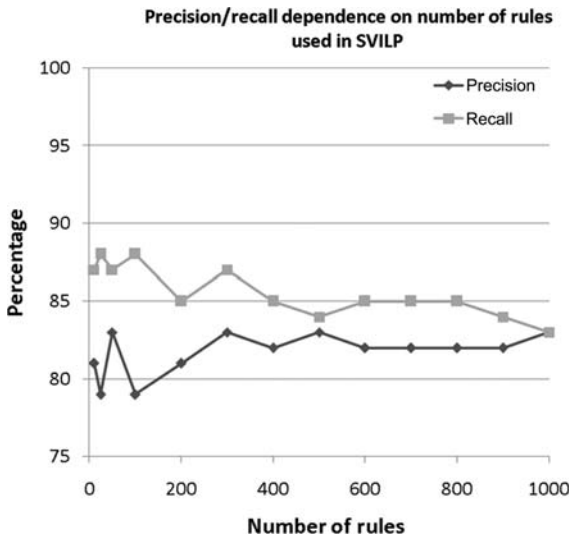

Fig. 1. Graph depicting how performance of the SVILP system was affected by the number of rules used to form the attribute vector. Data shown are performance on the independent 5-fold optimisation set.

chose to use the 300 highest compression rules applied to the full 20-fold cross-validation. For each of the 300 ILP-derived rules, a given binding pocket was represented as a binary attribute vector where an attribute is '1' if the rule is true for that example or zero otherwise. Thus, compared with the simple SVM approach, a far smaller representation of the pocket in binary form is used. The SVILP system achieved 83% and 87% precision and recall, respectively, which constitutes a further marginal improvement over either ILP or SVM alone.

Frequency-based SVILP

One of the advantages of the ILP approach is the potential insight one can gain from the comprehensible rules it generates. We investigated where in the binding pockets the ILP patterns were matching. To our surprise, we discovered that there were often many instances of residue triplets matching a given rule for positive examples. The standard SVILP approach assumes a simple binary truth function for a rule, i.e. the triplet is either present or absent in the pocket. However, we discovered that there is more information present in the form of the rule matching frequency. This can be understood by considering the idea of 'patches' of some biophysical property.

Consider a triplet 'hydrophobic, polar, polar' with particular dimensions, as described above. If the binding pocket contains a hydrophobic patch and two polar patches matching the required dimensions, then there are multiple instances where the ILP-derived rule is true because of the flexibility of the triangle. Almost none of the ILP-derived rules is completely free of false positives matches. However, upon analysing the rule matching frequencies, we discovered that false positive examples usually match only once or twice for a given rule, whereas true positives may match over 10–15 instances. This observation led us to investigate a modified form of SVILP using frequency information.

Instead of forming an attribute vector for an example based on a binary decision of whether a rule is matched or not, we used the frequency of a match as an attribute (see the Methods section). This resulted in superior performance to the SVILP approach while using a radically reduced number of features (Table I, Fig. 2). The *f*-SVILP system achieved 90% and 86% precision and recall, respectively.

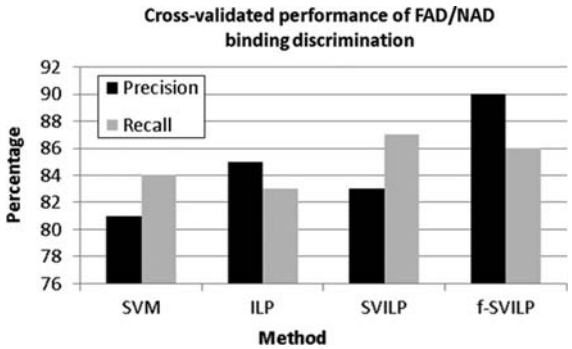

Fig. 2. Graph depicting the performance of each of the methods on the FAD/NAD discrimination problem. Precision and recall have been calculated over the 20-fold cross-validation. Precision is defined as tp/(tp + fp) and recall is defined as tp/(tp + fn), where tp are true positives, fp the false positives and fn the false negatives. Full data are presented in Table I.

Biophysical basis of the discovered rules

We investigated in detail the behaviour of the most compressive rules to gain some insight into their potential biophysical basis.

We investigated the binding pocket of a hydroxynitrile lyase from almond (PDB code 1JU2) (Dreveny *et al*., 2001). This protein was chosen as it had a large number of high frequency 'hits' by the ILP rules. Every amino acid in the binding pocket of 1JU2 was assigned a number based on how many times it played a part in an instantiation of an ILP rule (the top 10 most compressive ILP rules from a randomly selected leave-one-out replicate were used). The residues of the binding pocket were then ranked by this number. Starting with the most frequently involved residue in the pocket, residues were cumulatively added to a list until 50% of all instantiations were covered. This is intended to capture those residues most important for accurate discrimination. We call these residues 'hubs' because of their frequent role as vertices in the set of triangles in the pocket.

By colouring the residues in a binding pocket according to the frequency with which they take part in matches to the top 10 ILP rules, one can visualise the regions of the pocket that are most important in the ability of ILP to discriminate between FAD and NAD pockets. One may then investigate whether these features may be relevant to the way the protein discriminates between these two cofactors. As can be seen from Fig. 3, there are three distinct red patches in the pocket. Because these hubs form patches of biophysical properties, we hypothesised that there may be some connection between their geometry and the conformational flexibility of the ligand.

To investigate the possible relationship between the residues frequently found as vertices in the ILP-derived rules and the conformational flexibility of the ligand, we used AutoDock 4.0 (Morris *et al*., 1998) to generate 200 docked conformations of an FAD molecule in the binding pocket of 1JU2. We then examined the intra-atom distances across the ensemble of 200 docked FAD molecules. The variance in

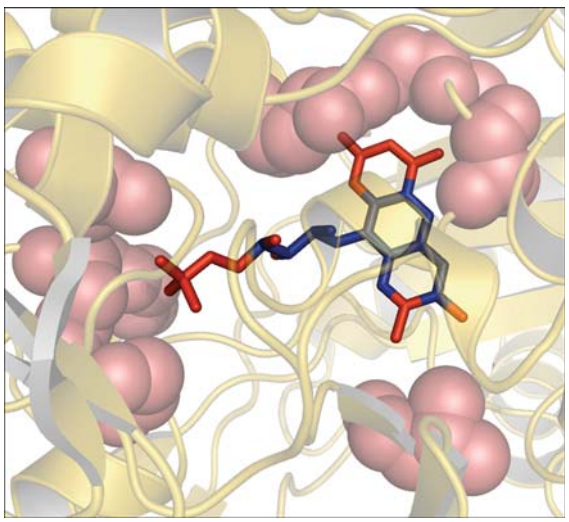

Fig. 3. Cartoon representation of FAD (blue and red sticks) inside the binding pocket of protein 1JU2. Red sticks indicate regions of the FAD that exhibit higher conformationally flexibility across 200 computational docking simulations. Pink spheres indicate atoms within amino acid residues that account for 50% of the instantiations (proofs) of the top 10 ILP rules.

distance for each FAD atom pair was summed up at each atom of the molecule and the atoms ranked by their summed variances. We have designated as 'variable' those atoms in the top 50% of this ranked list of variance values. Colouring this molecule in accordance with this conformational flexibility shows three regions of relatively high mobility (Fig. 3).

Visually, one can appreciate that the hubs discovered by ILP match well in three-dimensional space with the regions of higher mobility in the ligand.

Discussion

We began by developing a representation of ligand-binding pockets in proteins by considering spatially separated triplets of amino acids and their biophysical properties. Using ILP to automatically learn rules governing specificity for FAD or NAD permitted good discrimination on a cross-validated test set using only a handful of rules. A pure SVM approach was also considered and performed comparably but required 2100 attributes and is not amenable to understanding and insight. We then applied the SVILP approach to the problem and achieved a marginal increase in performance while maintaining a considerable reduction in the number of attributes (300).

Further investigation led to the discovery of valuable latent information in the frequency of occurrence of rule-based features within the binding pockets. This inspired the development of a new approach to SVILP which we call *f*-SVILP and this in turn produced a substantial improvement in precision (90%) and recall (86%) with only a handful of rules (20). Further investigation of the geometric distribution of the triplet patterns within the binding pocket has led to the suggestion that these patterns are related to the conformational flexibility of the ligand within the binding pocket. Tentative evidence to this effect has been shown by the good correlation between the proximity of the most flexible regions of the ligand and those residues of the binding pocket most frequently involved in the automatically discovered rules.

Unfortunately, due to the almost complete lack of FAD-binding proteins in the unbound state (only one protein was found in the PDB), it is not possible to test the methodology on unbound structures on this data set. Nevertheless, because of the nature of the rules, we can place strict limits on the degree of conformational freedom the current system permits. First, the system will be immune to alterations in side-chain conformations as these are completely ignored in this analysis. This is also a feature of value when handling predicted protein structures where side-chain placement may be highly inaccurate. Further, any pair of Cβ atoms can vary in their mutual distance by ± 2.5 Å which represents a considerable conformational change. Multiple changes larger than this are expected to reduce performance.

This initial study lays the groundwork for a general method of predicting ligand specificity solely from knowledge of the backbone structure and composition of the binding pocket. The search space of all possible residue triplets with all the combinations of possible representations of the biophysical attributes of the amino acids is too large for a conventional analysis. Instead, by relying on the relational power of ILP, this space can be searched efficiently to generate compact, comprehensible rules in a relatively short time ($\sim$24 CPU hours in this work). In this way, ILP is being used to tackle the common problem in machine learning of 'feature extraction'. The discovery of the latent information in the rule matching frequency has permitted the development of a new machine learning technique that surpasses the already high accuracy of the other methods while using only a fraction of the number of rules/attributes. This greatly eases the process of analysing the potential biophysical relevance of the rules.

This work demonstrates the feasibility of discovering simple rules governing protein–ligand interactions. Future work is aimed at extending these rules to cover all major ligand types. This can be implemented using a set of pairwise discriminators as developed above for every pair of ligand binding site classes, or in a 'one vs. all' setting where rules are learnt that discriminate one ligand site from all others. Such rules would not only permit the development of a general protein–ligand binding predictor, but would also have implications for protein function design. The top 1000 discovered rules, protein datasets and progol code are available at: http://www.sbg.bio.ic.ac.uk/svilp_ligand/. A tutorial and software for implementing SVILP is available at http://www.doc.ic.ac.uk/~cjz/SVILP/.

Acknowledgements

The authors would like to thank Dr Huma Lodhi and Dr Jianzhong Chen for helpful discussions.

Funding

This work was supported by the Biotechnology and Biological Sciences Research Council (BB/E000940/1 to L.A.K.).

References

Amini,A., Lodhi,H., Muggleton,S.H. and Sternberg,M.J.E. (2007a) *J. Chem. Inf. Model.*, **47**, 998–1006.

Amini,A., Shrimpton,P.J., Muggleton,S.H. and Sternberg,M.J.E. (2007b) *Proteins: Struct. Funct. Bioinf.*, **69**, 823–831.

Baker,D. and Sali,A. (2001) *Science*, **294**, 93–96.

Brenner,S.E. (2001) *Nat. Rev. Genet.*, **2**, 801–809.

Burley,S.K., Almo,S.C., Bonanno,J.B., Capel,M., Chance,M.R., Gaasterland,T., Lin,D., Sali,A., Studier,F.W. and Swaminathan,S. (1999) *Nat. Genet.*, **23**, 151–157.

Cannon,E.O., Amini,A., Bender,A., Sternberg,M.J.E., Muggleton,S.H., Glen,R.C. and Mitchell,J.B.O. (2007) *J. Comput. Aided Mol. Des.*, **21**, 269–280.

Cootes,A., Muggleton,S.H. and Sternberg,M.J.E. (2003) *J. Mol. Biol.*, **330**, 839–850.

Dodson,G. and Wlodawer,A. (1998) *Trends Biochem. Sci.*, **23**, 347–352.

Dreveny,I., Gruber,K., Glieder,A., Thompson,A. and Kratky,C. (2001) *Structure*, **9**, 803–815.

Fetrow,J.S. and Skolnick,J. (1998) *J. Mol. Biol.*, **281**, 949–968.

Fetrow,J.S., Godzik,A. and Skolnick,J. (1998) *J. Mol. Biol.*, **282**, 703–711.

Finn,P., Muggleton,S., Page,D. and Srinivasan,A. (1998) *Mach. Learn.*, **30**, 241–270.

Gibrat,J.F., Madej,T. and Bryant,S.H. (1996) *Curr. Opin. Struct. Biol.*, **6**, 377–385.

Henikoff,S., Henikoff,J.G. and Pietrokovski,S. (1999) *Bioinformatics*, **15**, 471–479.

Holm,L. and Sander,C. (1995) *Trends Biochem. Sci.*, **20**, 478–480.

Huang,J.Y. and Brutlag,D.L. (2001) *Nucleic Acids Res.*, **29**, 202–204.

Hu,L., Benson,M.L., Smith,R.D., Lerner,M.G. and Carlson,H.A. (2005) *Proteins: Struct. Funct. Bioinf.*, **60**, 333–340.

Joachims,T. (1999) In Schölkopf,B., Burges,C. and Smola,A. (eds), *Advances in Kernel Methods—Support Vector Learning.* MIT-Press, Cambridge, MA.

King,R.D., Muggleton,S.H., Srinivasan,A. and Sternberg,M.J.E. (1996) *Proc. Natl Acad. Sci. USA*, **93**, 438–442.
King,R.D., Whelan,K.E., Jones,F.M., Reiser,P.K.G., Bryant,C.H., Muggleton,S.H., Kell,D.B. and Oliver,S.G. (2004) *Nature*, **427**, 247–252.
Laskowski,R.A., Watson,J.D. and Thornton,J.M. (2005) *J. Mol. Biol.*, **351**, 614–626.
Morris,G.M., Goodsell,D.S., Halliday,R.S., Huey,R., Hart,W.E., Belew,R.K. and Olson,A.J. (1998) *J. Comput. Chem.*, **19**, 1639–1662.
Muggleton,S.H. and De Raedt,L. (1994) *J. Logic Programming*, **19**, 629–679.
Muggleton,S.H., Lodhi,H., Amini,A. and Sternberg,M.J.E. (2005) *Proceedings of the 8th International Conference on Discovery Science*, LNAI 3735, pp. 163–175.
Orengo,C.A., Jones,D.T. and Thornton,J.M. (1994) *Nature*, **372**, 631–634.
Sadowski,M.I. and Jones,D.T. (in press) *Curr. Opin. Struct. Biol.*, doi:10.1016/j.sbi.2009.03.008
Skolnick,J., Fetrow,J.S. and Kolinski,A. (2000) *Nat. Biotechnol.*, **18**, 283–287.
Stark,A. and Russell,R.B. (2003) *Nucleic Acids Res.*, **31**, 3341–3344.
Sternberg,M.J.E. and Muggleton,S.H. (2003) *Qsar Comb. Sci*, **22**, 527–532.
Tamaddoni-Nezhad,A., Chaleil,R., Kakas,A. and Muggleton,S.H. (2006) *Mach. Learn.*, **64**, 209–230.
Vapnik,V. (1995) *The Nature of Statistical Learning Theory*, Springer-Verlag, New York.
Wallace,A.C., Laskowski,R.A. and Thornton,J.M. (1996) *Protein Sci.*, **5**, 1001–1013.
Wallace,A.C., Borkakoti,N. and Thornton,J.M. (1997) *Protein Sci.*, **6**, 2308–2323.
Zhao,S., Morris,G.M., Olson,A.J. and Goodsell,D.S. (2001) *J. Mol. Biol*, **314**, 1245–1255.

Received June 9, 2009; revised June 9, 2009; accepted June 11, 2009

Edited by Samy Meroueh