doi:10.1016/S0022-2836(03)00628-4 J. Mol. Biol. (2003) 330, 771–783

JMB

Available online at www.sciencedirect.com

SCIENCE DIRECT®

Distinguishing Enzyme Structures from Non-enzymes Without Alignments

Paul D. Dobson and Andrew J. Doig*

Department of Biomolecular Sciences, UMIST, P.O. Box 88 Manchester M60 1QD, UK

The ability to predict protein function from structure is becoming increasingly important as the number of structures resolved is growing more rapidly than our capacity to study function. Current methods for predicting protein function are mostly reliant on identifying a similar protein of known function. For proteins that are highly dissimilar or are only similar to proteins also lacking functional annotations, these methods fail. Here, we show that protein function can be predicted as enzymatic or not without resorting to alignments. We describe 1178 high-resolution proteins in a structurally non-redundant subset of the Protein Data Bank using simple features such as secondary-structure content, amino acid propensities, surface properties and ligands. The subset is split into two functional groupings, enzymes and non-enzymes. We use the support vector machine-learning algorithm to develop models that are capable of assigning the protein class. Validation of the method shows that the function can be predicted to an accuracy of 77% using 52 features to describe each protein. An adaptive search of possible subsets of features produces a simplified model based on 36 features that predicts at an accuracy of 80%. We compare the method to sequence-based methods that also avoid calculating alignments and predict a recently released set of unrelated proteins. The most useful features for distinguishing enzymes from non-enzymes are secondary-structure content, amino acid frequencies, number of disulphide bonds and size of the largest cleft. This method is applicable to any structure as it does not require the identification of sequence or structural similarity to a protein of known function.

Keywords: protein function prediction; structure; enzyme; machine learning; structural genomics

*Corresponding author

Introduction

We aim to demonstrate that protein function can be predicted as enzymatic or not without resorting to alignments. Protein function prediction methods are important as international structural genomics initiatives are expected to generate thousands of protein structures in the next decade. The capacity of laboratories studying protein function is not sufficient to keep pace with the number of structures being released, with the consequence that many new structures lack functional annotations. Predictions can help guide the activities of these laboratories towards functionally more important proteins. The main approaches to *in silico* protein function prediction from structure are neatly summarised by Sung-Ho Kim.[1] They are assignment of a function from a similar fold, sequence, structure or active site of previously known function, assignment from a structure with a common ligand and *ab initio* function prediction (implying a method that does not work by comparison with another protein of known function).

The most common methods rely on identifying similarity to a protein of known function and transferring that function. Sequence alignments are identified using approaches such as BLAST[2] or FASTA.[3] The power of PSI-BLAST[4] has permitted the detection of sequence similarities that infer homology down to below 20%. Even when the likes of PSI-BLAST fail, the sequence can still yield useful information in the form of sequence motifs, which can be identified using PRINTS,[5] BLOCKS,[6]

Abbreviations used: EC, Enzyme Commission; CE, Combinatorial Extension.

E-mail address of the corresponding author: andrew.doig@umist.ac.uk

PROSITE[7] and other similar tools. Using predicted secondary structures to assign fold class can expand the information content of a sequence still further, since fold classes are often associated with a particular set of functions.[8]

The next logical step after using predicted structure is to use real structure. As structure is more highly conserved than sequence, it is often possible to detect similarities that are beyond the reach of even the most sophisticated sequence alignment algorithms. Structural similarity is detected using tools such as Combinatorial Extension[9] and VAST,[10] which map structures onto each other. Incomplete structural alignments can still suggest fold class. A problem encountered when identifying similar folds is that there may not be one specific function associated with a fold, making choosing the correct one non-trivial. The TIM barrel fold is known to be involved in at least 18 different enzymatic processes[1] and while this does give a narrowing of the number of possible functions to assign, the precise function remains unknown.

Transferring function from a protein that shares a ligand is a method that can give variable results if not tempered with some biochemical knowledge. For example, possession of NADH suggests an oxidoreductase enzyme. Functionally unimportant ligands may be shared by many structures, but to say that these proteins share a common function would be far from accurate. Ligand data can be used in conjunction with data concerning the immediate protein environment that binds the ligand. Binding-site correspondence is a strong indicator of functional similarity,[11] as is the case with the correspondence of the near-identical catalytic triads in the active sites of trypsins and subtilisins,[12] two evolutionarily distant but functionally similar types of protein. The utility of this approach is demonstrated by the ProCat database.[13]

For sequences and structures that are highly similar, the reliability of the predicted function is good, though in a recent study it has been shown to be less than previously thought.[14] For pair-wise sequence alignments above 50%, less than 30% share exact EC numbers. This suggests the level of sequence/structure conservation that implies function conservation is much lower than believed formerly and demonstrates the pressing need for protein function prediction methods that are not dependent upon tools that detect alignments.

Non-alignment-based function predictions have been made using many different techniques. Text data mining of scientific literature[15] uses the information in scientific abstracts to assign subcellular localisations, which can be used as an indicator of function. Amino acid compositions have been used to predict localisation.[16,17] The Rosetta Stone[18] method allows function predictions to be made for proteins that do not align to a protein of known function by examining gene fusions. If the protein aligns to part of a fused protein and the part of the fused protein it does not align to matches a protein of known function, that function can be transferred to the original protein. Phylogenetic profiling[19] functionally relates proteins with similar profiles. The gene neighbour method uses the observation that if the genes that encode two proteins are close on a chromosome, the proteins tend to be functionally related.[20,21] Neural networks have been used to combine predicted post-translational modifications into sophisticated systems capable of predicting subcellular location and function.[22]

While similarity-based methods do provide the most precise and dependable means of function prediction, in many cases it is apparent that they are heavily reliant on being able to identify highly similar proteins of known function. With one of the principal objectives of the structural genomics initiatives being the elucidation of structures from the more sparsely populated regions of fold space, the problem of not finding a similar protein of known function is more likely to occur. A method suggested by Stawiski *et al*.[23] that lies between a similarity-based approach and an *ab initio* method, is based on the observation that proteins of similar function often use basic structural features in a similar manner. For example, they note that proteases often have smaller than average surface areas and higher C^{α} densities. Similarly, *O*-glycosidases[24] deviate from the norm in terms of features such as the surface roughness (or fractal dimension). Features identified as being indicative of a certain function permit the construction of machine-learning-based classification schemes that allow function predictions for novel proteins without resorting to conventional similarity-based methods. The broad structural similarities that characterise a functional class of proteins extend beyond the reach of structural alignments, yet it has been shown that they can be used for protein function prediction. Here, we demonstrate a method of identifying protein function as enzymatic or not without resorting to alignments to proteins of known function. To do this, we describe each protein in a non-redundant subset of the Protein Data Bank[25] in terms of simple features such as residue preference, residue surface fractions, secondary structure fractions, disulphide bonds, size of the largest surface pocket and presence of ligands. As we are demonstrating a method for use when alignment methods do not yield results, we restrict ourselves to features that do not rely on alignments. As such, our method is for use when alignment methods fail. Histograms illustrate that for some features the distributions of enzymes and non-enzymes are different. In order to utilise these differences we combine the data into a predictive model using the support vector machine technique. Adaptive programming is used to find a more optimal subset of features, giving a greater predictive accuracy whilst simultaneously simplifying the model. We validate these models by leave-out analyses and predicting a set of unrelated proteins submitted to the Protein Data Bank since the training set was compiled.

Table 1. All features and a more optimal subset

Fraction of total residues	**Fraction of surface area**	**Heterogens**
ALA	***ALA***	***ATP***
ARG	ARG	***FAD***
ASN	***ASN***	***NAD***
ASP	ASP	Calcium
CYS	***CYS***	Copper
GLN	***GLN***	Heme
GLU	GLU	***Iron (not heme)***
GLY	GLY	
HIS	HIS	**Fraction secondary structure**
ILE	***ILE***	***Helix***
LEU	***LEU***	***Sheet***
LYS	***LYS***	Turn
MET	***MET***	
PHE	***PHE***	**Other**
PRO	PRO	***Size of CASTp largest pocket***
SER	***SER***	
THR	THR	***Disulphide Bond***
TRP	TRP	
TYR	***TYR***	
VAL	***VAL***	

Greyed features are excluded from the more optimal subset

Features used to describe the difference between structures of enzymes and non-enzymes. Bold and italicised features are those selected by the adaptive search of possible subsets as part of the most optimal subset identified.

Using the same approach, we investigate the utility of models built only using amino acid propensities. Being easily calculable from sequence, this provides a method for predicting the function of proteins that cannot be aligned to a protein of known function, even if we do not have a structure. We also make a comparison to the ProtFun enzyme/non-enzyme methods described by Brunak *et al*.[22]

Results

The support vector machine works by deducing the globally optimal position of a hyperplane separating the distribution of two classes of points scattered in a multi-dimensional space. The number of features used to describe the position of points determines the dimensionality of that hyperspace. The 52 features used to describe each protein are shown in Table 1. All features are easily calculable from any protein structure. No feature is based on mapping sequence or structure onto a known protein, so the model can be said to be truly independent of alignment-based techniques.

Heterogen data is presented to the machine in binary form (1 for present, 0 for absent; Table 2).

Table 2. Ligand groups in the dataset

Hetero group	Enzyme	Non-enzyme
Metals and metal-containing compounds		
Calcium	46	23
Cobalt	1	0
Copper	2	1
Heme	21	15
Iron (not heme)	4	1
Manganese	0	0
Nickel	1	0
Zinc	0	1
Cofactors		
ATP	33	6
FAD	40	0
NAD	26	0

Numbers of each hetero group type in the dataset and each class. Metals present only once in the dataset were not used in the training set.

Some metal ions are present only once in the dataset, in which case the feature was not included in the set of features on which the machine was trained.

Results from the leave-out analyses of prediction accuracy are shown in Table 3. These data are based on a model generated using a radial basis kernel function. Linear and polynomial kernels perform less well (not shown).

When using all 52 features, the results indicate a predictive accuracy for the set of features around 76–77%. Random performance based purely on the size of classes could give, at best, an accuracy of 58.7% by predicting "enzyme" every time. The subset selection strategy selects 36 features from the total 52. The increase in accuracy is approximately 3–4% to 80%. Bold and italicised features in Table 1 show the subset that gives the results in Table 3. On a set of 52 unrelated (by Combinatorial Extension) proteins submitted to the Protein Data Bank since the training set was compiled, this 36 feature model predicts with an accuracy of 79%. Three unrelated proteins with no functional assignment in the Protein Data Bank were predicted as follows (confidence in parentheses): 1JAL non-enzyme (1.081), 1NY1 non-enzyme (−0.166), 1ON0 enzyme (0.548).

Table 3. Average percentage accuracies and average standard errors of models built on all features and the optimal subset of 36 features in Table 1

	Leave-*x*-out			
Number of features	1%	5%	10%	20%
All (52)	77.16	76.87	76.86	76.17
Standard error	–	1.24	1.23	1.29
Subset (36)	80.14	80.26	80.17	80.43
Standard error	–	1.28	1.24	1.31

Comparison of the prediction accuracy of models generated from all features and the more optimal subset of features generated by an adaptive search of possible subsets. The accuracy increase is approximately 3–4%.

Table 4 shows that certain features are selected more than others. Possibly as a result of its low abundance at protein surfaces, the surface fraction of tryptophan is never used. Ligand data are used only rarely (calcium is never used). This is probably related to the sparseness of heterogens in the dataset. Other features, particularly secondary-structure contents (Figure 2) and the volume of the largest pocket (Figure 3) are selected frequently.

The confidence of a support vector machine prediction is related to the distance from the hyperplane. Larger distances equate to higher confidence. Figure 1 illustrates how the 36 feature models in a leave-one-out analysis were predicted. On enzymes, the method is 89.7% accurate. For non-enzymes, the accuracy is 68.6%.

Models trained only on amino acid propensities give an indication of how accurate function prediction can be from sequence when a similar protein of known function cannot be detected (Table 5).

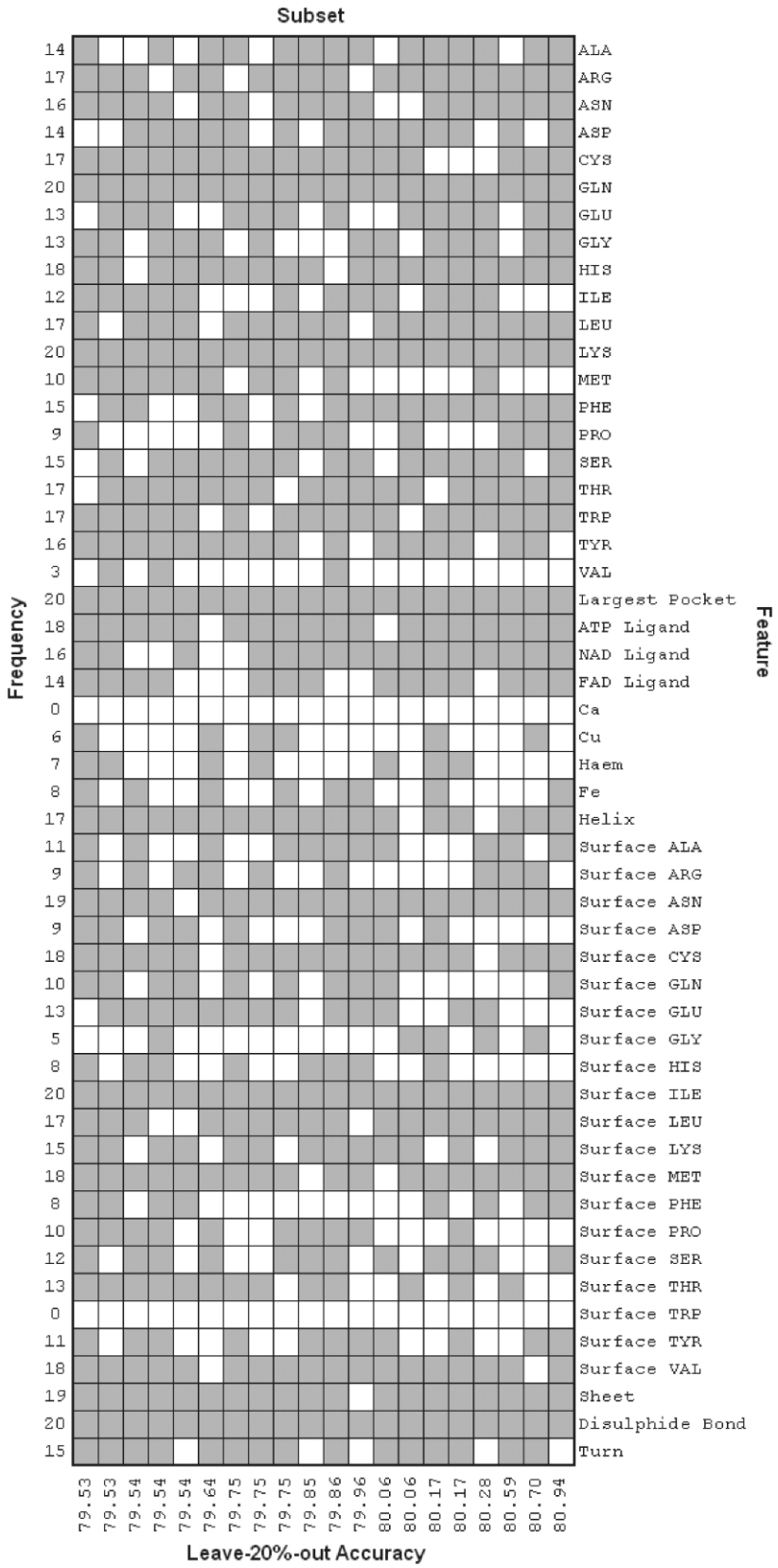
Subsets contain only the features that combine to more optimally describe the difference between classes using the support vector machine. To visualise this, histograms can be used to depict the underlying probability distribution of classes for a particular feature. Figures 2–5 show typical histograms.

Correlations between class distributions for certain features, such as largest pocket volume, helix and sheet contents, are low. Most other features have distributions that tend to differ only slightly for the two classes. However, the support vector machine combines individual features into a multi-dimensional joint probability distribution, so it is the manner in which feature distributions interact that is important. It is for this reason that wrapper methods were preferred to filters for subset selection (see Methods). A pre-filter method based on a correlation coefficient would have resulted in a less optimal performance.

For α-helix fraction (Figure 2) it is evident that there are large differences between enzymes and non-enzymes, and the coiled-coil state that is observed in some DNA-binding proteins is depicted at the upper end of the scale by a slight increase in the non-enzyme density. Most other histograms approximately fit to standard statistical distributions such as normal, Poisson, etc. as typified by Figures 4 and 5. Figure 3 shows the volume of the largest surface pocket. Above 1000 Å^3, enzymes are more prevalent, so a protein with a large surface pocket is more likely to be enzymatic. Conversely, and this may be more relevant, proteins with small surface pockets tend not to show enzyme activity.

The ProtFun[22] prediction servers also predict whether a protein is an enzyme. These servers work entirely on features calculated from protein sequence. Like the method presented here, they are not reliant on alignments. We submitted sequences from all the proteins in the dataset to ProtFun analysis. The results indicate that on this set the ProtFun approach is 75.21% accurate.

Table 4. Twenty subsets chosen by adaptive searching



The composition of 20 models selected by the adaptive programming regime. Grey shaded squares indicate that a feature is present in a subset. The number of times a feature was selected over 20 runs is shown on the left. Percentage accuracies for each model are shown along the bottom.

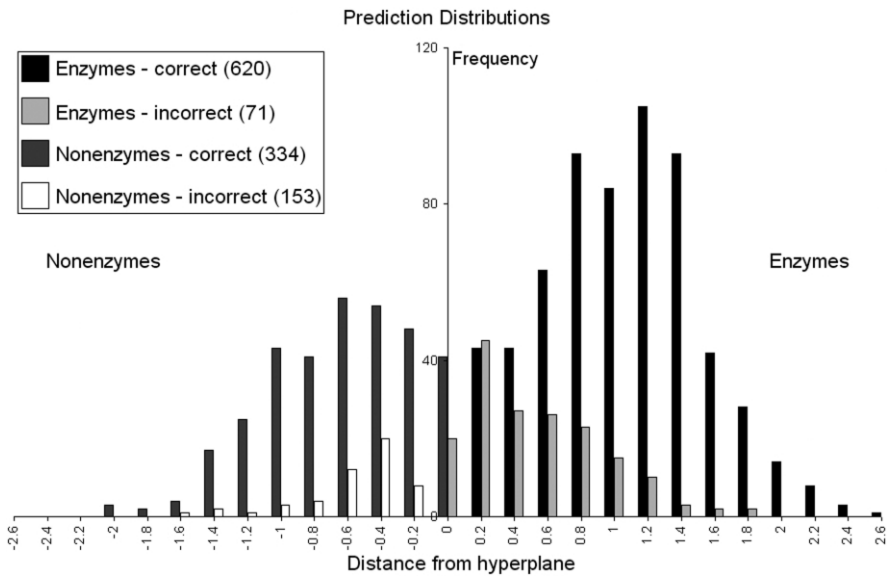


Figure 1. Prediction confidence distributions. Distance from the hyperplane is related to the confidence of a prediction. The distributions of predictions for an 80.9% (36 feature) model are shown.

Discussion

It is apparent that there is a need for methods to predict protein function when conventional approaches do not yield results. We demonstrate the utility of representing proteins not in terms of the precise locations of residues, but by using simple features such as residue preference, secondary structure, surface features and ligands. When these data are combined using the support vector machine approach, a model is built that can predict the class of a novel protein as enzymatic or not to an accuracy of approximately 77%. This accuracy is based on a model built upon 52 features. A subset of these features selected by a basic adaptive program gives a much simpler model, capable of predicting with an increased accuracy of 80%. Figure 1 suggests that the model works by defining what constitutes an enzyme rather than a non-enzyme, as it predicts enzymes with much greater accuracy. This makes sense, as enzymes are truly a class of proteins, with some similar properties such as having active sites, whereas non-enzymes are merely all those proteins that are not enzymes that have been accumulated into something more artificial and vague. One might expect that with greater diversity in the non-enzyme class, a model that isolates enzymes from the rest would be easier to derive. It is also apparent from Figure 1 that high-confidence predictions are rarely incorrect.

The method chosen for modelling the data and making predictions is the support vector machine,[26,27] a machine-learning tool for two-class classification. The goal of classification tools is to find a rule that best maps each member of training set S (described by the properties X) to the correct classification Y. Each point in the input space is described by the vector (x_i, y_i) :

$$x_i \in X \subset \Re^N$$

$$y_i \in Y = \{+1, -1\}$$

The support vector machine is based on separating the N-dimensional distribution of points in a training set. A hyper-plane is fit to the input space in a manner that globally optimally separates the two user-defined classes. The orientation of a test sample relative to the hyper-plane gives the predicted class.

Recently, the popularity of the support vector machine has increased dramatically. It has been shown to frequently out-perform previous classification favourites such as neural networks. There are two major features of the support vector machine that make it capable of such high performance, the kernel function and structural risk minimisation.

It is evident that for most real problems the optimal separating hyper-plane will not fit linearly between the two classes without significant error, though a non-linear separation may well be feasible. The kernel function permits the mapping of data in the input space into a higher-dimensional feature space where non-linear mappings become linearly possible. The choice of kernel function determines the nature of the mapping to the feature space.

Perhaps the most critical component of the support vector machine is the manner in which risk is minimised. The concept of risk can be defined as the number of errors the model makes. It is apparent that a good model should make as few errors as possible. Traditionally, risk is minimised by fitting a model to the training set so that it can

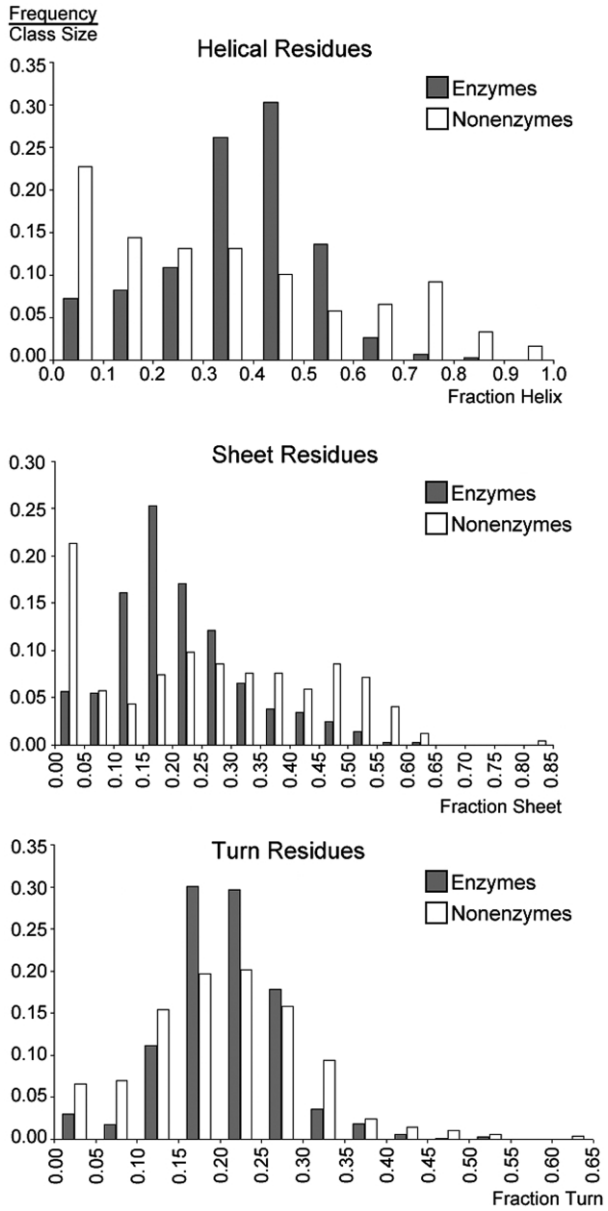


Figure 2. Histograms to illustrate differences in secondary structure contents.

predict the training set as accurately as possible. This is known as empirical risk minimisation. However, while the model may be able to predict the training set almost perfectly, it may not perform so accurately on data unlike what it has been trained on. The model is said to have lost the ability to generalise. If the training set can be guaranteed to represent everything the model could be asked to predict, then the problem would not be so great, but this situation is rare. The result of empirical risk minimisation is often a model with poor predictive abilities as a consequence of describing the training set too well. In this situation, we say that the model is over-fit to the data. Risk is minimised differently by the support vector machine. Structural risk minimisation uses a measure of the number of different models implementable in the training set to control the point at which a model can be said to be over-fit to the training data. In this way, generalisation ability can be controlled. A rule can be found that balances the generalisation ability with the empirical risk minimisation method to ensure that the prediction performance is optimal on the training set without being over-fit to it.

The adaptive search of possible subsets was preferred to filter methods that quantify the difference between classes for each feature, as it explores the joint probability distributions of possible subsets in the context of the learning algorithm. This means that the subset selection strategy investigates features as they interact, rather than in isolation. The final subset selected by the adaptive programming search (what remains upon completion of the adaptive process) is not always the same. Local performance maxima within the set can be responsible for this. Without an exhaustive search it is difficult to show that the adaptive programming approach finds the most optimal subset from the 2^{52} possibilities. However, it consistently finds simple subsets capable of predicting with greater accuracy without sacrificing generalisation performance (by our measure of generalisation, in which the subset is prevented from over-fitting by restricting the tolerance of variation in leave-out results so that any subset of features incapable of predicting well on all fractions of the dataset is not permitted).

Table 4 illustrates that certain features are selected consistently, such as the secondary-structure fractions (Figure 2), largest pocket volume (Figure 3), fractions of phenylalanine (Figure 4), fractions of surface asparagine and isoleucine, and ATP, NAD and FAD. Some of these more frequently selected features have class distributions that differ sufficiently to permit slightly improved (compared to random) function predictions without other features. For example, the volume of the largest pocket is a feature that one might expect to differ greatly between classes and, indeed, it shows little correlation between enzymes and non-enzymes. Enzymes tend to have large surface pockets containing their active sites. Therefore, a protein without a large surface pocket is probably not an enzyme. This seems to be true, as the majority of enzymes have a largest pocket volume above 1000 $Å^3$. Above this volume there are still non-enzymes as a result of those proteins that require large surface pockets for binding but have no enzymatic role (for example, a DNA-binding protein such as a transcription factor).

The presence of any of the coenzymes ATP, FAD and NAD in a structure suggests strongly that the protein is an enzyme. ATP is found in some non-enzyme structures, which is to be expected, as it is involved in almost all processes that involve energy transfer. Despite this, in the majority of cases it is still indicative of an enzyme.

Another feature that exhibits poor correlation is helix content. A cursory glance at the helix content

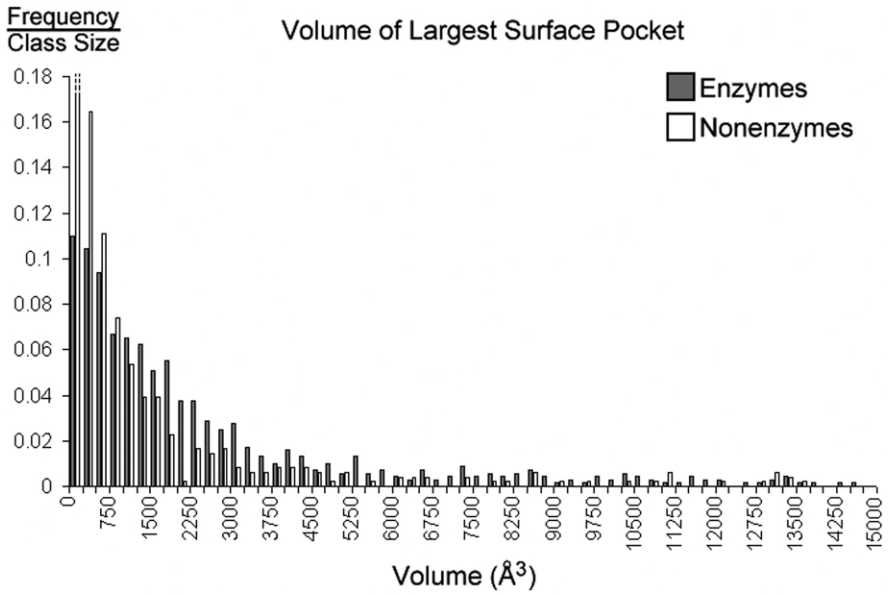


Figure 3. Differences in the volume of the largest surface pocket (Å^3). Pockets of volume above 1000 Å^3 suggest the protein is an enzyme. (The non-enzyme value in the leftmost bin, excluded here for clarity, is 0.32).

distributions suggest that alone it should permit a better than random prediction, as proteins with less than 30% helix tend to be non-enzymes, between 30–60% enzymes tend to dominate, and above 60% non-enzymes are more prevalent. Similar information can be gleaned from the sheet fractions, with a significant proportion of non-enzymes having no β-sheet. Between 10% and 30%, enzymes are more common than non-enzymes. Greater than 30% sheet content suggests non-enzymes. That secondary structure contents have such discriminating power is intriguing and opens up avenues of research employing secondary-structure predictions or experimental data.

For most features, there is little apparent difference between class probability distributions that suggests they will have any discriminating power. Yet it is clear that these features do have some utility, as they are chosen consistently by adaptive searching. Biological interpretations for some of the more frequently selected features can be proposed, though these are difficult to verify. Many residue types are known to be associated with certain processes linked to function and/or subcellular location, such as the role of asparagine in N-linked glycosylation (in Asn-X-Ser/Thr, X ≠ Pro). This may explain why surface asparagine is a useful feature, as certain classes of proteins may require glycosylation (such as certain extracellular proteins that require enhanced solubility).

A set of residues that accounts for only a small fraction of the total surface includes cysteine, isoleucine, leucine, methionine and valine. Surface fractions of these hydrophobic residues are selected frequently. One possible explanation for this is that a residue type with a preference for being buried, and so normally present at the surface at only low levels, is significant when it is found unburied. That is, something that introduces instability, like having a solvent-accessible aliphatic residue, would not happen without purpose. This suggests involvement in the protein's function or cellular location. Patches of hydrophobic residues at the surface probably occur in proteins that interact with other proteins, making complex formation energetically more feasible.

The high frequency of selection of the histidine fraction feature is interesting. It is not a highly abundant residue at the surface, yet it has the interesting property of having a pK_a of 6–7, which makes it amenable to mediating proton transfer. This is a common process in many enzyme mechanisms and offers an explanation for the higher frequency at which it occurs in the enzyme class.

Several features are selected very rarely, if at all. Removal of such features allows for a much simpler model, as the level of abstraction by the kernel function from the input space to the feature space is much lower in the absence of noisy or useless features. In combination, features can cooperate to form a joint probability distribution

Table 5. Average percentage accuracies and standard errors of models built on amino acid frequencies

	Leave-*x*-out			
Features	1%	5%	10%	20%
20	75.38	75.00	74.83	74.98
SE	–	1.33	1.37	1.35

The performance of models built on all amino acid frequencies and the associated standard errors.

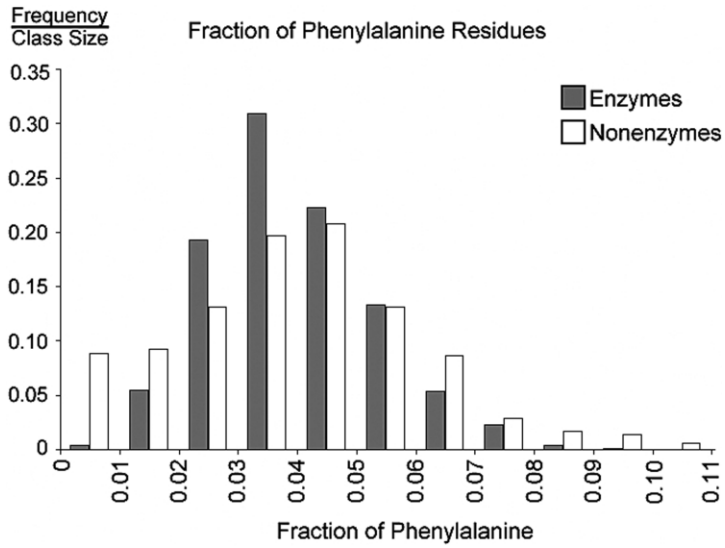


Figure 4. Example histogram. The differences in fractions of phenylalanine between enzymes and non-enzymes.

that lends itself well to partitioning, so too a feature can act to disrupt this and force a less well-fit model.

Sparse features, such as the metal ion data, seem to contribute little. It is surprising that some types of metal are present so infrequently in this dataset. Calcium is not present infrequently, nor is it distributed equally between classes. However, it does not get used in any of the models generated.

To summarise the feature selections, it is clear that some features are useful for predicting protein function in isolation, as the distributions differ greatly between classes. In combination, this discriminatory power is heightened. Features that seem to correlate well across the two classes of function can bring something to the predictive method that is not immediately apparent from their distributions. It is worth noting that the radial basis function kernel performs better than linear and polynomial kernels. This is indicative of the high complexity of the dataset distribution. Evidently, different sub-groupings of each class exist in pockets throughout the total distribution, forming discontinuous clusters of data. Of the three kernels used, the radial basis function is most capable of competently mapping complex data such as this into higher dimensions, where the problem becomes linearly tractable.

The construction of the dataset is intended to avoid the problem of over-representation in the Protein Data Bank, yet it still suffers the problem of under-representation. For example, it is notoriously difficult to obtain structures for membrane proteins and, as a consequence, there are very few deposited in the Protein Data Bank. Another source of bias stems from the dataset containing only X-ray crystal structures (as a consequence of the programmes used to calculate certain features being intolerant of the ambiguity that is often seen with NMR structures). The structures that are not amenable to crystallisation are therefore potentially under-represented. The consequence of these biases is that models generated may be less capable of predicting functions for

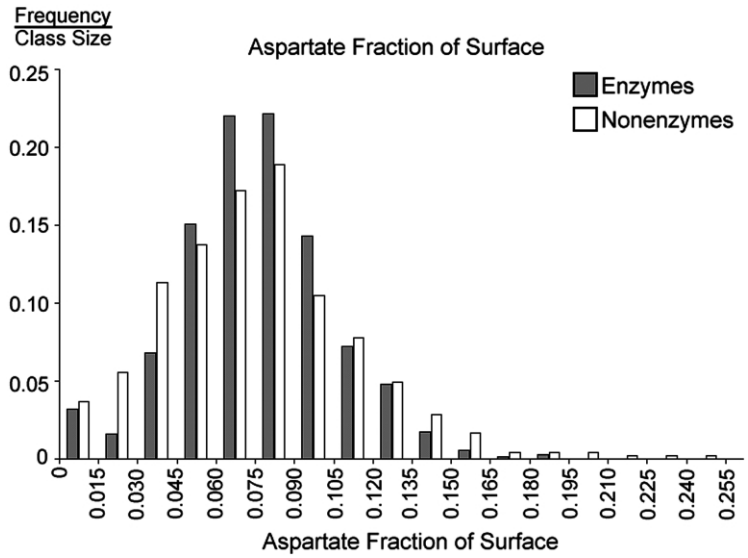


Figure 5. Example histogram. The differences in surface fractions of aspartate between classes.

under-represented proteins. As under-represented classes have structures solved, the method should easily be able to accommodate this extra information.

There is an uneven distribution of enzyme classes in the PDB (with hydrolases being by far the largest class and ligases much the smallest). This is reflected in the dataset, so one may suspect that the model may focus on predicting certain classes more than others. However, errors in a self-consistent test (using a model to predict the set on which it was trained) range only from 4–11% over the six top-level enzyme classifications, suggesting that the model is not overly directed towards any particular class. For non-enzymes, there is no equivalent classification scheme applicable here, but from functional annotations within the PDB files no bias is immediately obvious.

Without structures it is still possible to make function predictions. Purely working from amino acid frequencies, models can be built in the same way that are capable of predicting to approximately 75% accuracy. It seems prudent to validate this result in future work by using a larger, more comprehensive dataset based on a non-redundant sequence dataset (here the results are based on the structural dataset and so are subject to the biases listed above). With a dataset that covers a more diverse selection of proteins the problem changes, so perhaps the accuracies shown herein should be treated with a degree of caution. From this dataset it is clear that predictions to some above-random extent are certainly possible using only amino acid frequencies.

A comparison is made with the ProtFun method described by Brunak *et al*.[22] by running our dataset through their algorithms. This is not really a comparison on an even footing, as there are more data available to us in a structure, but more data available to ProtFun in that they have a larger dataset that may contain sequences similar to sequences in our dataset. However, the comparison highlights a greater accuracy for the method herein, and goes some way to vindicating the inclusion of structure data in protein function predictions.

We have demonstrated the utility of representing proteins in terms of simple features as a method of describing the similarities between proteins with a common function. Using this information, we have been able to construct models that can predict protein function as enzymatic or not to an accuracy of 80%. We are currently in the process of setting up a server that will take as input a PDB file†, extract data for the features used in our best performing model and make predictions.

† http://wolf.bms.umist.ac.uk/~pad/predict/enon.html

Methods

Dataset construction

The dataset consists of X-ray crystal structures with a resolution of less than or equal to 2.5 Å and an *R*-factor of 0.25 or better. A structurally non-redundant representation of the Protein Data Bank provides a firmer grounding for validating results as prediction accuracies are artificially high with a redundant dataset (it is easier to make a correct prediction for an object if the model is built upon data that is essentially the same). Removing similarity also avoids the problem of biases in the PDB that are the result of certain research areas producing more structures than others.

The Combinatorial Extension[9] methods implemented at UCSC provide structural alignment *Z*-scores for all chains in the PDB against all others, or approximations through a representative structure. This information is sufficient to build the set of structures required. The authors of CE propose a biological interpretation of the *Z*-scores. Scores of 4.5 and above suggest the same protein family, scores of 4.0–4.5 are indicative of the same protein superfamily and/or fold, while scores of 3.5–4.0 imply a possible biologically interesting similarity. In this cull of the PDB, no chain in any protein structurally aligns to any other chain in the dataset with a *Z*-score of 3.5 or above outside of its parent structure. Aligning to another chain in a parent structure is permitted, as redundancy is only an issue in leave-out testing, so similarity between chains that are always left out together does not bias results.

The dataset is split into 691 enzymes and 487 non-enzymes on the basis of EC number, annotations in the PDB and Medline abstracts. Ten proteins were excluded from the total cull due to incomplete function annotation. The dataset is provided as a supplement.

Features for model building

Easily computed features that differ between classes are used to describe each protein. Residue preference is calculated as the number of residues of each type divided by the total number of residues in the protein. Secondary structure content is calculated similarly working from STRIDE[28] assignments. The helix content is derived from the total amount of the structure assigned as α, 3_{10} or π helix. Sheet and turn fractions are calculated similarly. Disulphide bond numbers are derived from the SSBOND section of the PDB file. The size of the largest pocket is the molecular surface[29] volume in $Å^3$, as calculated by the cavity detection algorithm CASTp.[30] Surface properties are calculated from NACCESS,‡ with the fraction of the total surface attributable to each residue type being features.

Heterogen data are taken from the PDB file and presented to the machine in binary form (1 for present, 0 for absent). When ligands are used in crystallisation, often an analogue or a derivative of a structure is used for technical reasons. This has the potential to make data very sparse. For example, it is desirable to cluster adenosine triphosphate with the structures for adenosine mono- and diphosphate. There are methods available

‡ Hubbard, S. J. & Thornton, J. M. (1993). "NACCESS," Department of Biochemistry and Molecular Biology, University College London.

that cluster similar hetero groups together, yet they often group molecules that are clearly different. After investigating the utility of certain clustering schemes, it was decided that only a very simple and severe classification scheme would classify rigorously enough to avoid error. The feature ATP contains all structures that contain one of the following HET codes: AMP, A, ADP, ATP. These are codes for adenosine mono-, di- and triphosphate. Similarly, the feature FAD consists of proteins that contain either FAD or FDA. We consider the oxidised and reduced forms of FAD as equivalent. The NAD feature uses the HET codes NAD, NAH, NAP and NDP. The NAD feature treats NAD and NADP in both oxidised and reduced states as equivalent. In order to find equivalent HET codes the "stereo smiles" utility in the Macromolecular Structure Database was used.[31] While this gives some scope for detecting when analogues have been used, it does not cover the whole range of alternative hetero groups that could be used as analogues. A looser clustering of hetero groups may cover these, but would not provide the level of specificity necessary to prevent noise being introduced. In terms of restricting noise, the more austere scheme detailed above performs best. Completeness is sacrificed to preserve accuracy.

All non-binary features are normalised over the range to take a value between 0 and 1, as recommended with support vector machines.

The support vector machine

The implementation of the support vector machine used here is SVMLight†.[32] The kernel functions used were linear, polynomial and radial basis function. Other than varying the kernel function, the algorithm was run on default settings.

Performance analysis: leave-out-analysis

In order to validate the method, some measure of accuracy is required. A simple but reliable method of assessing accuracy is the leave-out method of cross validation.[33] From a set S a subset of size x is excluded, with the remaining $S-x$ members forming the training set. The model built from this training set is used to predict the classes of members of the excluded subset x. Each of the S/x subsets is left out and predicted in turn. The sum of the accuracy on each subset divided by the number of subsets gives an estimate of total accuracy. At its lower extremity, x can be a portion of size containing only one member (a leave-one-out analysis), though for larger datasets this is often computationally expensive and a larger value of x is preferred. The results from different values of x are not always consistent. There is debate concerning the value x should take. Here, results are presented for x being one protein, 5%, 10% and 20% of the set. The percentage leave-out analyses are each repeated ten times with different partitions to ensure that any result is not purely the consequence of a fortuitous splitting of the dataset. Mean performance and standard error are averaged over the ten partitions.

† http://svmlight.joachims.org

Performance analysis: newly released PDB files

Since the dataset was compiled the PDB has released many protein structures. The majority of these proteins are related to existing structures in some way. By using the same Z-score criterion as before, 56 new proteins that are not related to the dataset were identified. These split into 29 enzymes and 23 non-enzymes, with 1L4X being discarded due to it being a *de novo* designed peptide. Three further structures annotated as having no known function were 1JAL, 1NY1 and 1ON0.

Performance analysis: comparison to ProtFun[22]

The ProtFun servers predict protein function from sequence without resorting to alignments. A sequence submitted in FASTA format is processed to generate features for a neural network-based prediction. Using the probabilities generated, we compared our method. When a structure contained multiple chains, the average probability was used.

Feature subset selection by adaptive programming

In the presence of two models of differing complexity and approximately equivalent performance there is no reason to presume that the more complex model is any more valid than the simpler model (by the principle of Occam's Razor).[34] Choosing a more relevant subset of features leads to a simpler model that can often be of greater accuracy. This can be due to noisy features that introduce unhelpful disturbances to the joint probability distribution and make fitting the hyperplane more complex.

Methods for choosing a subset of features exist in two basic forms.[35] Filter methods rely on pre-processing the data to find features that differ between the two classes. They employ techniques that quantify the extent to which the two class distributions for a feature correlate (with poorly correlating features being useful for distinguishing between classes). Filter methods take little account of how the distributions of features interact in multidimensional space. They are more suited to selection of subsets from a very large number of features, when the computational expense of wrapper methods can prove prohibitive. Wrapper methods use the learning algorithm itself to search for a subset of features. Here, the learning algorithm is used as the selection criterion in a simple adaptive programming search of possible feature subsets.[36] There are 2^{52} possible subsets to choose from 52 features, making an exhaustive search impossible. Adaptive programming is well suited to querying large search spaces. A basic adaptive programme randomly generates solutions to a problem and then uses some measure of fitness to select the parents of the next generation of possible solutions. These parents then reproduce with mutations. Here, a simple adaptive programme starts from a random subset of features. Offspring from this parent are generated by introducing random mutations at a frequency of 10% per reproduction (that is, five changes are made, as this is approximately 10% of 52. It is permitted for a feature to change and change back). On the population size reaching 15, the leave-20%-out analysis forms the basis of the scoring system that is used to select the model with greatest accuracy. Here, there is a risk that the subset begins to describe the problem only for the dataset and in so doing loses the ability to generalise. If the

accuracy on each left-out set is similar, then the subset of features is capable of describing the difference between classes irrespective of the subset of data it is trained on. If the average prediction accuracy is high, but the standard deviation of accuracies over all left-out sets is also high, then it can be said that the model does not generalise so well. Low performance on some left-out sets is indicative of a subset of features that cannot describe the difference between classes for all data it could be asked to predict. The selection criteria for the adaptive programming regime are therefore based on the leave-20%-out estimated average performance, and on the standard deviation of these results. The maximum tolerated standard deviation is that generated by the whole set of features. A subset capable of predicting with greater accuracy than the rest of its generation, but with a standard deviation that is lower than or equivalent to the ancestor of all generations, can be said to have specialised no more than the common ancestor as a consequence of subset selection and so is considered more fit. For example, if the common ancestor predicts at 75% accuracy with a standard deviation of 2.5%, a subset that performs with $>75\%$ accuracy and $\leq 2.5\%$ is more fit.

After selection, a precise copy of the parent is always maintained in the next generation to avoid backwards steps. This subset gives rise to the next generation. After a maximum of 150 generations, the algorithm ceases. Repetition of the algorithm suggests the existence of different subsets of approximately equivalent performance, which is indicative of the adaptive programme following different trajectories to multiple local maxima. The local maximum with best performance is the best model.

Acknowledgements

This work was funded by a BBSRC Engineering and Biological Systems committee studentship. We thank Ben Stapley for helpful discussions and Kristoffer Rapacki of the Center for Biological Sequence Analysis, Technical University of Denmark for assistance with the ProtFun results.

References

1. Thornton, J. M. (2001). Structural genomics takes off. *Trends Biochem. Sci.* **26**, 88–89.
2. Altschul, S. F., Gish, W., Miller, W., Myers, E. W. & Lipman, D. (1990). Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410.
3. Pearson, W. R. & Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proc. Natl Acad. Sci. USA*, **85**, 2444–2448.
4. Altschul, S. F., Madden, T. L., Schaffer, A. A., Zhang, J., Zhang, Z., Miller, W. & Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucl. Acids Res.* **25**, 3389–3402.
5. Attwood, T. K. (2002). The PRINTS database: a resource for identification of protein families. *Brief. Bioinform.* **3**, 252–263.
6. Henikoff, S., Henikoff, J. G. & Pietrokovski, S. (1999). Blocksredundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics*, **15**, 471–479.
7. Falquet, L., Pagni, M., Bucher, P., Hulo, N., Sigrist, C. J., Hofmann, K. & Bairoch, A. (2002). The PROSITE database, its status in 2002. *Nucl. Acids Res.* **30**, 235–238.
8. Rice, D. & Eisenberg, D. (1997). A 3D–1D substitution matrix for protein fold recognition that includes predicted secondary structure of the sequence. *J. Mol. Biol.* **267**, 1026–1038.
9. Shindyalov, I. N. & Bourne, P. E. (1998). Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng.* **11**, 739–747.
10. Madej, T., Gibrat, J.-F. & Bryant, S. H. (1995). Threading a database of protein cores. *Proteins: Struct. Funct. Genet.* **23**, 356–369.
11. Russell, R. B., Sasieni, P. D. & Sternberg, M. J. E. (1998). Supersites within superfolds. Binding site similarity in the absence of homology. *J. Mol. Biol.* **282**, 903–918.
12. Wallace, A. C., Laskowski, R. & Thornton, J. M. (1996). Derivation of 3D coordinate templates for searching structural databases: application to the Ser-His-Asp catalytic triads of the serine proteinases and lipases. *Protein Sci.* **5**, 1001–1013.
13. Wallace, A. C., Borkakoti, N. & Thornton, J. M. (1997). TESS: a geometric hashing algorithm for deriving 3D coordinate templates for searching structural databases: application to enzyme active sites. *Protein Sci.* **6**, 2308–2323.
14. Rost, B. (2002). Enzyme function less conserved than anticipated. *J. Mol. Biol.* **318**, 595–608.
15. Stapley, B. J., Kelley, L. A. & Sternberg, M. J. E. (2001). Predicting the subcellular location of proteins from text using support vector machines. In *Proceedings of the Pacific Symposium on Biocomputing 2002, Kauai, Hawaii* (Altman, R. B., Dunker, A. K., Hunter, L., Lauderdale, K. & Klein, T. E., eds), World Scientific, Singapore.
16. Cai, Y., Liu, X. & Chou, K. (2002). Artificial neural network model for predicting protein subcellular location. *Comput. Chem.* **26**, 179–182.
17. Chou, K. & Elrod, D. (1999). Protein subcellular location prediction. *Protein Eng.* **12**, 107–118.
18. Marcotte, E. M., Pellegrini, M., Ng, H., Rice, D. W., Yeates, T. O. & Eisenberg, D. (1999). Detecting protein function and protein–protein interactions from genome sequences. *Science*, **285**, 751–753.
19. Pellegrini, M., Marcotte, E. M., Thompson, M. J., Eisenberg, D. & Yeates, T. O. (1999). Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proc. Natl Acad. Sci. USA*, **96**, 4285–4288.
20. Dandekar, T., Snel, B., Huynen, M. & Bork, P. (1999). Conservation of gene order: a fingerprint of proteins that physically interact. *Trends Biochem. Sci.* **23**, 324–328.
21. Overbeek, R., Fonstein, M., D'Souza, M., Pusch, G. D. & Maltsev, N. (1999). The use of gene clusters to infer functional coupling. *Proc. Natl Acad. Sci. USA*, **96**, 2896–2901.
22. Jensen, L. J., Gupta, R., Blom, N., Devos, D., Tamames, J., Kesmir, C. *et al.* (2002). Prediction of human protein function from post-translational modifications and localization features. *J. Mol. Biol.* **319**, 1257–1265.
23. Stawiski, E. W., Baucom, A. E., Lohr, S. C. & Gregoret, L. M. (2000). Predicting protein function

from structure: unique structural features of proteases. *Proc. Natl Acad. Sci. USA*, **97**, 3954–3958.
24. Stawiski, E. W., Mandel-Guttfreund, Y., Lowenthal, A. C. & Gregoret, L. M. (2001). Progress in predicting protein function from structure: unique features of *O*-glycosidases. In *Proceedings of the Pacific Symposium on Biocomputing 2002, Kauai, Hawaii* (Altman, R. B., Dunker, A. K., Hunter, L., Lauderdale, K. & Klein, T. E., eds), World Scientific, Singapore.
25. Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H. *et al.* (2000). The Protein Data Bank. *Nucl. Acids Res.* **28**, 235–242.
26. Burges, C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining Knowledge Discov.* **2**, 121–167.
27. Vapnik, V. (1999). *The Nature of Statistical Learning Theory*, Springer, New York.
28. Frishman, D. & Argos, P. (1995). Knowledge-based protein secondary structure assignment. *Proteins: Struct. Funct. Genet.* **23**, 566–579.
29. Connolly, M. L. (1983). Analytical molecular surface calculation. *J. Appl. Crystallog.* **16**, 548–558.
30. Liang, J., Edelsbrunner, H. & Woodward, C. (1998). Anatomy of protein pockets and cavities: measurement of binding site geometry and implications for ligand design. *Protein Sci.* **7**, 1884–1897.
31. Boutselakis, H., Copeland, J., Dimitropoulos, D., Fillon, J., Golovin, A., Henrick, K. *et al.* (2003). E-MSD: the European Bioinformatics Institute macromolecular structure database. *Nucl. Acids. Res.* **31**, 458–462.
32. Joachims, T. (1998). Text categorization with support vector machines: learning with many relevant features. In *Proceedings of ECML-98, 10th European Conference on Machine Learning, Chemnitz, Germany* (Nédellec, C. & Rouveirol, C., eds), Springer, Berlin.
33. Bishop, C. M. (1995). Cross validation. In *Neural Networks for Pattern Recognition*, sect. 9.8.1. pp. 372–375, Oxford University Press, Oxford, UK.
34. John, G. H., Kohavi, R. & Pfleger, K. (1994). Irrelevant features and the subset selection problem. In *Machine Learning: Proceedings of the Eleventh International Conference* (Cohen, W. W. & Hirsh, H., eds), Rutgers University, New Brunswick, NJ.
35. Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T. & Vapnik, V. (2000). Feature selection for support vector machines. *NIPS*, **13**, 668–674.
36. Holland, J. (1992). *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge.

Edited by J. Thornton

(Received 30 January 2003; received in revised form 28 April 2003; accepted 9 May 2003)

Supplementary Material comprising lists of PDB codes for the proteins used here is available on Science Direct