Mining Protein Structure Data

José Santos[1], Pedro Barahona[2], Ludwig Krippahl[2]

[1] Department of Computing, Imperial College, London, United Kingdom
jcs06@doc.ic.ac.uk
[2] Department of Computing, FCT-UNL, Lisbon, Portugal
{pb, ludi}@di.fct.unl.pt

Abstract. This paper describes the application of machine learning algorithms to the discovery of knowledge in a protein structure database. The problem addressed is the determination of the solvent exposure of each amino acid residue, using different levels of exposed surface to define exposure. First we introduce the baseline classifier which achieves good prediction results despite only taking into account the amino acid type. Then we explain how we gathered and processed the data and built our classifier to improve the baseline prediction. Finally we test and compare several classifiers (e.g. Neural Networks, C5.0, CART and Chaid), and parameters (level of information per amino acid, SCOP class of protein, sliding window from the current amino acid) that might influence the prediction accuracy. We conclude by showing our models present a modest but statistically significant improvement over the baseline classifier's accuracy.

Keywords: Amino acid Relative Solvent Accessibility, Protein Structure Prediction, Bioinformatics, Data Mining

1 Introduction

The function of a protein is by large determined by its surface and the interactions it can establish with other molecules (e.g. other proteins, drugs or other metabolites). More specifically, such interactions are highly dependent on the shape (3D structure) of the proteins and their ligands, since the interactions are established by relative weak forces (hydrogen bridges, electrostatic forces or free energy) that require relatively large contact areas with conformant shapes (key and lock effect). As important as it may be, shape alone is not sufficient to explain protein interaction, and the physicochemical properties of the surface also play an important role in such interactions. Hence, the importance of identifying the specific amino-acid residues that are exposed in the protein surface, since the surface properties are obtained by some aggregation of the individual properties of the amino-acids (such as polarity hydropathy).

Since protein structure is difficult to predict, many studies [1,2,3,4] try to circumvent this difficulty by aiming at identifying which amino-acid residues are exposed and contribute to protein surface properties, starting with (known) primary structure of the protein, supported as much as possible on (predicted) information about the secondary structure of the protein. This fact justifies that such studies are usually concerned with levels of exposure high enough to influence surface properties, and the studies address the residues for which at least 20 or 25% of their individual surface is exposed at the protein surface.

Our motivation for the study presented in this paper is slightly different. The approach we have been adopting for the determination of protein structure from inter-atomic distances provided by NMR studies, adopted hybrid constraint propagation and local search methods typical in constraint programming applications. In the first phase, the possible positions of the (centre of the) atoms, i.e. their domains, is reduced until an approximate solution is obtained. The second phase adopts a local search optimisation, by firstly adjusting the dihedral angles of the flexible chemical bonds to this approximate solution, and then improving it by small variations of these angles in order to better satisfy the distance and other geometrical constraints of the problem.

The first phase of this approach is thus very important, given the limitations of local search to efficiently exploit large portions of the search space (typical local search approaches, including the simulated annealing meta-heuristics, usually take a few hours in proteins for which NMR studies are applicable (say, up to two hundred residues) [11] . Good approximations are thus quite important for the success of our approach. Although we have been obtaining relatively good results (a few Angstroms of RMSD), [6], these are not yet as good as the before mentioned alternatives.

We have thus attempted, in addition to improved propagation techniques [10], to understand the importance of good heuristics for narrowing the domains of the atoms centres, and an exhaustive study has shown that finding a heuristics that fails less than 20% of the time, would significantly improve the final RMSD obtained [5]. The same study has also shown that a heuristic that is based only on geometric parameters (for example, the intersection volume of atoms domains) is not sufficient.

This was then the main motivation for our concern on the prediction of the exposition of residues in the protein surface. If the residue is at the protein surface, the heuristics that are used in the first phase (constraint propagation) of our approach may allow us to select from the domain part that is closer to the surface. Moreover, this context justifies that we are not only interested in 20-25% of exposed areas, but on lower surface fractions (2% and 10%), which are more informative for the heuristics.

In this paper we present the data mining studies we have done on the protein structures available in the Protein Data Bank, in order to predict exposed surface areas (at 2%, 10%, 20%, 25% and 30%) of the amino acid residues. The paper is organised as follows. In section 2 we present the sources of biochemical information from we built our local database. In section 3 we present the methodology followed in our work, describe the baseline to measure the merit of our experiments, the parameters used for learning and the machine learning classifiers. In section 4 the results of the experiments are presented and its significance is tested. Finally in section 5 conclusions and comparison with related work are presented.

2 Sources of biochemical information

We obtained the data for this work from two public databases of biochemical information, the Protein Data Bank (PDB) [12] and the Structural Classification Of Proteins database (SCOP) [13]. To determine the exposed surface amino acid residues we used the Definition of Secondary Structure of Proteins (DSSP) algorithm [14], a standard method for assigning secondary structure elements. This section gives some details on these sources.

2.1 Protein Data Bank (PDB)

The PDB is the central depository for protein structures. With over 44 thousand structures, as of July 2007, it contains virtually all current data on protein structure, and so was the natural choice of a data source for this work. In addition, the PDB is updated weekly, constantly providing new data that can be used to test or validate data mining methods. Finally, the PDB provides a set of useful software tools that facilitate mirroring and processing the database.

2.2 Definition of Secondary Structure of Proteins (DSSP)

The DSSP algorithm is a standard method for identifying secondary structure elements in protein structures. However, the main use for DSSP in this work was to calculate the exposed surface of each amino acid residue in the protein structures, and, for consistency, also to calculate the total surface area of each amino acid in isolation.

The main reason for using DSSP is because it is used in most works on the exposed surface area of amino acid residues in proteins. Unfortunately, other authors often do not use the same method for determining the exposed surface of residues within the protein and the free amino acids, which makes it hard to compare results.

2.2 Structural Classification of Proteins (SCOP)

The SCOP database is one of the two main systems of structural classification, endorsed by the PDB, the other being the Class, Architecture, Topology and Homologous superfamily (CATH) [15]. The differences between the two are mostly due to different methods for defining protein domains, locally organized and often conserved parts of protein structures. Since our goal was to use these structural classifications to eliminate the redundancy of dealing with many similar structures, either system would do equally well, since both group similar structures together, even if the group classifications differ.

At the top of the SCOP hierarchy, the SCOP class, proteins are grouped by global features such as size (there is a special group for small proteins) or content of secondary structure elements (such as all alpha helix proteins). The second level, the fold, groups proteins by structural motifs and similarity to well conserved domains, such as globins or DNA-binding domains. The third level, the superfamily, groups proteins on their similarity to domains of specific proteins that serve as a template or centroid for the group. The fourth level, the family, groups protein structures by their structural homology. Proteins belonging to the same SCOP family are highly homologous.

3 Methodology

In this section we describe the methodology followed. As a data mining project we followed roughly the CRISP-DM Methodology. In the following sub sections we: 1) describe the collecting of the data and its preparation for the later steps; 2) explain how we computed the residue accessible area; 3) introduce the parameters used for learning; 4) describe the baseline classifier and how it is calculated; 5) explain the creation of train and test sets; 6) compare the accuracy of several classifiers and determine the most suitable to our task.

3.1 Preparing and cleaning the data

In April 2005 we downloaded all the biological units (about 30.000 by then, available here: ftp://ftp.rcsb.org/pub/pdb/data/biounit/coordinates/all) from the Protein Data Bank FTP server. Then we parsed the PDB files and imported them to a local relational database. The DSSP program was executed in these PDB files to generate exposed information on the amino acids. Finally, the SCOP table was also imported to our database.

There are several advantages in having the data in a relational database, the main one being ensuring data consistency. After loading the tables from the several sources, some inconsistencies appeared (e.g. PDBs that did not have a SCOP entry or vice-versa) which needed to be fixed (e.g. by eliminating such entries from the mining process). Another important advantage is the easiness of doing complex queries to explore the data, which is an important task in every mining project.

3.2 Calculating residue accessible area

The main aim of this paper is to predict the Residue Solvent Accessibility (RSA) exposure state. RSA is simply the ratio between Residue Exposed Area (REA) and Residue Total Area (RTA). REA varies with the position of the amino acid inside the protein (and is one of the outputs of DSSP) and RTA is roughly constant for each of the twenty amino acids types.

To calculate the residue total area (RTA) of an amino acid most papers in the literature use their surface areas available in several tables. However, each table has its own values. For instance, in papers [2,3,5], RTA values are taken from experimental data published in the 1970s or 1980s (e.g. [8]).

We have opted not to use any of those tables and calculate the RTA using also DSSP because all papers consensually calculate REA with DSSP. We calculate the RTA values by using DSSP in a special way: we isolate the amino acids of a pdb file in new pseudo pdb files with just that amino acid then, by running DSSP on those new small pseudo PDB files, we have, among other things, the exposed area to the solvent of the selected amino acids (the value is not constant because the amino acid might be taken from different conformations). This computation is performed for all amino acids taken from a large sample of randomly selected pdb files. The results are then averaged to find the solvent accessibility area for each amino acid type.

This average solvent accessibility area should be very close to the real residue area because the pseudo protein is now only the amino acid and so its area is totally exposed to the solvent. However, there are discrepancies with the values of the literature tables because those values are calculated considering the residue surface area and DSSP considers the residue accessible area to solvent (water). The residue surface area (RSA) is calculated by the total amino acid surface area, while the residue accessible area (RAA) is the surface the center a molecule of water, approximated by a sphere with a radius of 1.4 Angstroms, can reach.

There is a 0.99 correlation between RSA and RAA with RAA being roughly 1.66 times RSA (if using [8] as source). This relationship allows for a rough comparison between our PEA levels and the ones in other studies. For studies using RSA from [8] (e.g. [5]) our percentage of exposed area (PEA) roughly corresponds to their PEA/1.66. For instance, if they are using a 20% PEA value, we should compare it with our 12% PEA level (our closest PEA level is 10%).

3.3 Parameters for learning

The learning parameters used were basically static (i.e. constant) information about an amino acid. We divided this information in 3 classes: *Minimal*, *Simple* and *Complete*. In the *Minimal* case only the amino acid name and its total area were considered, in the *Simple* case six properties for each amino acid (Hydropathy, Polarity, H_Donor, H_Acceptor, Aromaticity, Charge at Ph7) were considered. Finally, in the *Complete* class we have 37 properties per amino acid (from [9]).

Other important parameter is the window size, that is, how many neighbors (to the left and right) of the current amino acid shall we consider (i.e. give information about). We used different window sizes of 0, 3, 6 and 10 amino acids. The level of information given about the neighbor amino acids is the same as for the current amino acid (i.e. either *Minimal*, *Simple* or *Complete* as well)

Another feature added is the protein size, measured in number of amino acids. The motivation is that this should help the mining algorithms as when the protein becomes bigger, the larger the ratio between its volume (a cubic function) and surface area (a quadratic function) becomes (i.e. in average there will be less exposed amino acids).

3.4 Baseline Classifier

In [3] Richardson and Barlow present a paper with a baseline for residue accessibility prediction models. Their baseline consists in assigning a residue into the particular exposure category in which it is most frequently found, not considering its local surrounding sequence. This baseline is the standard by which the literature measures its results. In our experiments we always present our models accuracy as improvement over this baseline classifier.

The baseline classifier works as follow: given a set of proteins as a dataset, it is split in training and test set as usual. In the training set we determine, for each amino acid, the frequency of its appearing in a certain class (i.e.: exposed/buried) for a certain degree of exposition (2%, 10%, 20%, 25% or 30%). The baseline model simple assigns an amino acid to the class where it appears more often (e.g.: if Alanine appears 60% of the time as exposed in the training set, our prediction for all Alanines in the test set is that they are exposed).

The baseline is hence dependent on the dataset and, more specially, on the cut off value for determining if a residue is exposed or buried. It can be considered as a special, very fast to calculate, classifier that has amino acid information *Minimal* and window size equal to zero. The table below represents the Baseline model, with the training being the consistent pdb biological units as of April 2005.

Table 1 Hydropathy, Average Exposed Area and inferred baseline classifier

Amino acid	**Hydro-pathy**	**Avg. Exp. Area**	**Percentage of burial**				
			2	**10**	**20**	**25**	**30**
Alanine	1.94	10.33%	0	0	1	1	1
Arginine	-80.00	23.20%	0	0	0	1	1
Asparagine	-9.70	20.59%	0	0	0	1	1
Aspartic	-80.65	22.44%	0	0	0	1	1
Cysteine	-1.24	5.38%	0	1	1	1	1
Glutamic acid	-79.12	26.91%	0	0	0	0	1
Glutamine	-9.38	22.40%	0	0	0	1	1
Glycine	0.00	12.02%	0	0	1	1	1
Histidine	-37.20	16.11%	0	0	1	1	1
Isoleucine	2.15	6.81%	0	1	1	1	1
Leucine	2.28	7.95%	0	1	1	1	1
Lysine	-69.24	29.94%	0	0	0	0	1
Methionine	-1.48	9.11%	0	1	1	1	1
Phenylalanine	-0.76	7.77%	0	1	1	1	1
Proline	0.00	18.85%	0	0	1	1	1
Serine	-5.06	15.92%	0	0	1	1	1
Threonine	-4.88	15.25%	0	0	1	1	1
Tryptophan	-5.88	9.74%	0	1	1	1	1
Tyrosine	-6.11	11.24%	0	0	1	1	1
Valine	1.99	7.48%	0	1	1	1	1

There is a correlation of -0.79 between the hydropath and average exposed area columns which indicate a strong relationship between these two attributes. The correlation is negative because for high values of average exposed area (ASA) correspond low values of hydropathy and vice-versa. This relationship will be explored further in section 4.

3.5 Train and Test Set division

A cluster of identical proteins can be defined as proteins having a high degree of homology (i.e. sharing a significant part of their amino acid sequence). In the Protein Data Bank the majority of proteins belong to large clusters while a minority belongs to small clusters (still with high homology). One of the problems this arises is that if we simply randomly chose train and test proteins it would be likely to have proteins from the same cluster in both the train and test set (which is not good if we want a general model). Other problem is that many small clusters would not be represented.

In order to overcome both problems we ensure only one protein represents a SCOP family (proteins in the same SCOP family have high homology). The protein that represents its SCOP family and which families belong to the train or test set is randomly chosen by a user defined seed. 2/3 of the proteins/families were used for training and the remaining 1/3 for testing purposes. There were 1742 families from all SCOP classes, with the 1180 of them having less than 10 proteins representing it.

Optionally, when generating the train and test sets, we can enforce the SCOP class of the families to be the same. This is interesting to have specific models per SCOP class and to test if that leads to an accuracy increase. In order to do an experiment we always generate (varying the random seed) five different pairs of train and test sets. The reported model's accuracies are the average accuracy on those five test sets.

3.6 Data Mining algorithms used

In this project we have used Clementine which is a commercial data mining framework from SPSS. It has a wide variety of machine learning algorithms but for this project we were only interested in the supervised learning classifiers, which are: C5.0, Neural Networks, Chaid, Classification and Regression Trees (CART). An important issue is to find which of these classifiers is more suitable for our problem.

Table 2 shows the results of executing the four classifiers and baseline in a specific scenario with window size 6 and amino acid information *Minimal*. The proteins used could belong to all SCOP classes (i.e. the model is not specific to a SCOP class).

Table 2 Accuracies for the different classifiers with Proteins from any SCOP Class, window size 6 and amino acid information *Minimal*

Classifier	Percentage of exposed area				
	2%	10%	20%	25%	30%
Baseline	75.59%	65.99%	70.31%	72.73%	77.60%
C 5.0	75.59%	69.03%	70.83%	73.44%	77.96%
Chaid	75.80%	68.95%	70.74%	73.36%	77.99%
CART	75.97%	68.88%	70.71%	73.36%	78.14%
N. Network	75.12%	67.91%	70.18%	72.80%	77.10%

Generating such tables for various window sizes but always with amino acid information *Minimal* takes some minutes (time to build the model is proportional to the window size and number of features per amino acid). However if we increase the amino acid information to *Simple* (or *Complete*) many hours are required. In order to be able to do the experiments efficiently we had to choose one classifier.

Although the gains are very modest (except for PEA 10) some improvements are statistically significant over the baseline. However, between the classifiers usually the differences are not statistically significant. To do experiments with more information per amino acid, detailed in section 4, much more time would be needed. Therefore, we decided to use only C5.0 for these experiments since it is the fastest classifier by far (2-4 times faster), is as good as (often even better than) the other classifiers and, very importantly, generates the model easier to understand by a human.

4 Experiments and Results

In this section we describe the experiments conducted and their results. Namely we measure the changes in the model accuracy by assessing the importance of 1) SCOP classes information, 2) the four amino acid window sizes (0, 3, 6 and 10), 3) the three levels of amino acid information (*Minimal*, *Simple* and *Complete*). In sub-section 4.3 we present the best prediction models and finally in 4.4 the robustness of the models is evaluated against a validation set.

4.1 Accuracies for different SCOP classes

One of the aims of this project was to test how the classifier accuracy varies for the different SCOP classes and if there is any gain in building specific models for some SCOP classes. We investigated only the four more populous classes (*a*, *b*, *c* and *d*) which, together, represent about 87% of all the proteins.

Besides these four specific models, two more models were developed. One, called *All*, which has exactly one protein representing each SCOP family. The other, called *Any*, which does not care about SCOP and randomly chooses proteins from the whole database. In no situation SCOP columns (class, fold, superfamily and family) were used as input for classifiers. Table 3 shows the baselines for the different SCOP classes at the different percentage of exposed areas (PEA).

Table 3 Baselines for the different Scop classes and percentage of exposed areas

Scop Class	Percentage of exposed area				
	2%	10%	20%	25%	30%
All	75,59%	65,99%	70,31%	72,73%	77,60%
Any	72,61%	65,01%	71,87%	75,49%	80,31%
A	77,52%	69,87%	70,89%	71,66%	75,55%
B	76,16%	66,29%	69,34%	72,45%	77,81%
C	70,59%	66,71%	73,15%	76,25%	80,67%
D	76,78%	67,85%	70,31%	72,30%	76,41%

Although apparently the baselines do not differ much between SCOP classes, some of the differences are statistically significant at 5% value. This gives evidence that the baseline is, at least for some percentages of exposed areas, SCOP class dependent. For instance, at 10% PEA, the baseline for class A is almost 70% when other baselines are about 66%, and at 2% PEA, the baseline for class C is only 70% when most other baselines are about 76%.

More interesting than analyzing the baselines are the classifier results for the different SCOP classes. Table 4 presents the improvement over the baseline, using C5.0 classifier, for amino acid window size 6 and amino acid information *Simple*.

Table 4 C 5.0 improvements over baseline with window 6 and information *Simple*

Scop Class	Percentage of exposed area				
	2%	10%	20%	25%	30%
All	0.96%	3.75%	0.95%	1.25%	0.92%
Any	6.09%	8.69%	4.24%	3.26%	1.85%
A	1.72%	3.18%	0.85%	0.96%	0.45%
B	1.31%	3.03%	0.62%	0.98%	0.72%
C	3.29%	3.67%	0.69%	0.44%	0.46%
D	0.29%	2.73%	0.47%	0.33%	0.39%

For SCOP class *Any* the improvement over the baseline is much larger than for any other class. This is not surprising since its train set contains randomly chosen proteins it is very likely that proteins in the test set belong to the same family as proteins in the training set and hence resulting in these extremely good results. We can also see that, except for 2% PEA, for all other SCOP models (*All, A, B, C* and *D*) the *All* classifier achieves the best accuracy.

We have built specific models for each of the Scop classes to test if they predicted better than the generic Scop *All* model predicts for the specific class. The results were similar. In fact, the Scop *All* model generally performed slightly better than the specific models for each class but the improvements were not statistically significant. Therefore, it is useless to build specific models for each SCOP class, so we will present only the results for the Scop *All* class.

4.2 Influence of amino acid window size and information level

The experiments we have performed considered four different window sizes: 0, 3, 6 and 10. A window size of *X*, considers the *X* neighboring amino acids to the left and the right of the current. It would be natural to think that the bigger the amino acid window size, the best the prediction of the burial status for the current, but that is not necessarily true. We also have three different levels of amino acid information: *Minimal*, *Simple* and *Complete*. The differences in these levels of information are explained above in subsection 3.3 but the number of features for each one is respectively 2, 8 and 37.

The time required for building the model varies significantly with the level of amino acid information and with the amino acid window size. Table 5 summarizes the times taken by the several classifiers for the distinct amino acid information levels and Table 6table 6 shows the increase in accuracy over the baseline gained by applying the C5.0 model for 10% PEA.

Table 5 Average time, in minutes, taken to build a SCOP All model with the four classifiers

Classifier	Amino acid Information level		
	Minimal	Simple	Complete
C 5.0 (Boost)	0.5	2.5	8.5
Chaid	1	3.5	22.5
Cart	3	5.5	56
Neural Network	3.5	5.5	62

Table 6 C5.0 model improvement over SCOP All, 10% PEA for the various levels of amino acid information and window sizes

Percentage of exposed area: 10% SCOP All (Baseline: 65.99%)				
Amino acid information	Amino acid window size			
	0	3	6	10
Minimal	2.81%	3.13%	3.04%	3.08%
Simple	2.59%	3.70%	3.75%	3.69%
Complete	2.75%	3.40%	3.47%	3.76%

The overall improvements are small, less than 3.80% in absolute value, showing that the baseline is a good estimate. Although the biggest improvement is for amino cid information *Complete* and a window size of 10, the best compromise is for amino acid information *Simple* and a window size of six.

Increasing the window size seems to help although, after window size six, the improvement almost ceases and sometimes even regresses. In the amino acid information side, adding more information seems to help from *Minimal* to *Complete* but not from *Simple* to *Complete*.

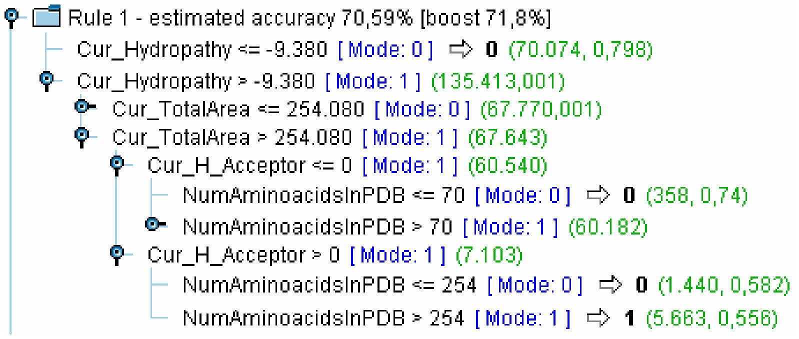
In order to clarify the relevance of the results, a student's t-test over the effect of the change in amino acid window size and information level was performed. The main conclusion is that increasing the window size from 0 to 3 is useful but from 3 to 6 and 6 to 10 is useless since there is no evidence the results are different (i.e. p-values are much larger than 5%). For other PEAs other than 10% sometimes it still slightly compensates increasing from 3 to 6 but never from 6 to 10.

We also note that there is a difference in passing from *Minimal* to *Simple*-which, looking at the accuracy table, is reflected in better predictions. On the other side, passing from *Simple* to *Complete* proves to be useless since the *p-values* are high. The patterns shown here for 10% PEA are valid for the other thresholds (2%, 20%, 25% and 30%). Their tables, with respective significance results, can be found in [1]. We can conclude that *Simple* information per amino acid is the most suitable value.

4.3 Best Model

From the results above, we conclude the best compromise between speed and accuracy is achieved with a window size of six and *Simple* amino acid information. In fact a window size of 10 and *Complete* information, besides taking much more time, might even lead to worse results (perhaps because the classifier algorithm gets puzzled with so much data). We also observed SCOP *All* model is the most generic classifier because it performs as well as the specific model for their own classes.

There are, in fact, five best models, one for each of the five PEA values (2, 10, 20, 25 and 30). They are, however, very similar - varying sometimes the values where to split the intervals and rarely the order of attributes- and hence showing one is enough for illustration purposes. Figure 1 shows the top excerpt of a C5.0 decision tree, as presented by Clementine, for 10% PEA with a window size six, amino acid information *Simple* and for SCOP *All*.



```
Rule 1 - estimated accuracy 70,59% [boost 71,8%]
  Cur_Hydropathy <= -9.380 [ Mode: 0 ] ⇨ 0 (70.074, 0,798)
  Cur_Hydropathy > -9.380 [ Mode: 1 ] (135.413,001)
    Cur_TotalArea <= 254.080 [ Mode: 0 ] (67.770,001)
    Cur_TotalArea > 254.080 [ Mode: 1 ] (67.643)
      Cur_H_Acceptor <= 0 [ Mode: 1 ] (60.540)
        NumAminoacidsInPDB <= 70 [ Mode: 0 ] ⇨ 0 (358, 0,74)
        NumAminoacidsInPDB > 70 [ Mode: 1 ] (60.182)
      Cur_H_Acceptor > 0 [ Mode: 1 ] (7.103)
        NumAminoacidsInPDB <= 254 [ Mode: 0 ] ⇨ 0 (1.440, 0,582)
        NumAminoacidsInPDB > 254 [ Mode: 1 ] ⇨ 1 (5.663, 0,556)
```

Figure 1 Top excerpt of C5 model for PEA 10, amino acid information *Simple*, window size 6 and SCOP *All*

We chose PEA 10 because it is where the gain over the baseline is more noticeable. We believe the more noticeable improvement is due to the specific cut off value making the baseline less accurate (rather than the model being particularly better).

The most important attribute to predict the buried status of the current amino acid is its Hydropathy (*Cur_Hydropathy* attribute). If it is lower than -9.380, C5.0 terminates immediately, answering the amino acid will be exposed (0), otherwise the default option is buried (1) but it might change depending on subsequent attributes. Hydropathy, being the most significant attribute, is precisely what we expected earlier from Table 1. A low hydropathy value means the amino acid is hydrophilic, while a high value means amino acid is hydrophobic, and hence there is a chemical attraction by the hydrophilic amino acids to the solvent and repulsion for the hydrophobic ones.

The second most important attribute is the total area of the amino acid. If the hydropathy of the current amino acid is higher than -9.380 and the total area is smaller than 254 the default prediction changes from 1 (buried) to 0 (exposed). Other important attribute is the number of amino acids in the current chain. The analysis continues for other attributes as depicted in the figure.

Another important consideration is that, in the excerpt of the decision tree presented, the only attributes in use are the ones about the current amino acid (the ones prefixed with *Cur*). The attributes regarding the neighboring amino acids are also used but have much less importance, appearing lower in the tree. However, as we have seen in the previous sub section 4.2, they are statistically significant and responsible for the improvement in accuracy of the classifier.

4.4 Validation Set Results

In the previous subsections we shown the results of the data mining algorithms applied to the test data. The main conclusions were to consider an amino acid window size of six, amino acid information *Simple*, C 5.0 classifier and SCOP *All*, therefore the experiments presented in this section were all performed with those parameters. We validate the best models by running them with new data, gathered more than a year after the one used for training and testing the models. This validation test is crucial to assess the models robustness built from the previous experiences.

There are two interesting validation experiments to perform. In the first experiment only proteins from new SCOP families (i.e. families not existing in the previous train or test data) are used. In this scenario we can evaluate how well the generated models for the training and test data have generalized for new families. Table 7 shows the result of applying the best model to only the new SCOP families in the validation data. Again only one chain was chosen to represent each family.

Table 7 Results of applying best model to new SCOP families in validation data

Classifier	Percentage of exposed area				
	2%	10%	20%	25%	30%
Baseline	76.35%	62.72%	70.63%	72.71%	76.66%
C50/All	77.41%	71.06%	71.43%	73.92%	77.50%
C50/Any	76.44%	69.63%	70.35%	73.21%	77.14%

Here we can see clearly that the C5.0/All model always predicts better than the C5.0/Any and that, except for the PEA 10%, the SCOP *Any* model barely surpasses the baseline. This is a clear indication that the *All* model has generalized very well and that the *Any* model has clearly over fitted not being much more useful than the baseline to predict the burial status of an amino acid.

In the second experiment we want to determine if a model built specifically for a certain class in the training set would outperform the generic SCOP *All* model on its own class. Table 8 compares the results of applying the generic SCOP *All* and *D* models, built with the training set, and applied to all chains that belong to new SCOP class *D* families, again ensuring only one chain per family.

Table 8 Results of applying best model to new SCOP class *D* families in validation data

Classifier	Percentage of exposed area				
	2%	10%	20%	25%	30%
Baseline	76.59%	68.02%	70.93%	72.65%	77.24%
C50/All	77.55%	70.85%	71.80%	73.32%	77.36%
C50/D	76.94%	70.18%	71.50%	73.36%	77.60%

The results show that the C5.0/All model accuracy is always better than the C5.0/D model (except for 30% PEA) and both are better than the Baseline. It might seem a little surprising that a generic model outperforms a model that has been trained specifically with chains from that class, but that is likely to be due to the knowledge gained by having seen proteins from different classes, as happens with C5.0/All, contains more information to predict the structure of new families than a model that only knows about the existence of a single class, as is the case of C5.0/D.

5 Conclusions

One of the main results of this paper is that our predictive residue solvent accessibility models present a small but statistically significant improvement over the very simple and fast to compute Baseline classifier. These results, with the needed adaptations, are in line with other published work [1,2,4], with the novelty of our work being a) the usage of lower percentage of exposed areas, b) much larger dataset (with thousands of proteins rather than a few hundred) set with a generic methodology to generate it, c) thorough study of the relative importance of parameters in model accuracy, e) proper validation of the models with newly discovered proteins.

In particular we have seen that, from the four classifiers analyzed (Chaid, Cart, Neural Networks and C5.0), although their results are in general not statistically different, C5.0 is much faster (by one or two orders of magnitude) than the others and has the added benefit that its model is easier to understand by a human.

Also, as previously shown, by learning with chains from different SCOP families (the *All* models), rather than by randomly choosing chains among the entire database (the *Any* models), the models generalize better. In addition, the *All* models perform as well in specific family data (e.g.: when all the chains are from a given SCOP class) as the specific models in the same data.

Regarding the importance of learning parameters in the model accuracy, there are statistically significant improvements in using more information about an amino acid rather than just its letter. However, it is useless to use too much information, a good compromise seems to be five or six attributes (like the *Simple* information described above). In particular, the most important attribute is the amino acid's hydropathy.

More sophisticated machine learning approaches such as Support Vector Machines (SVMs) were not used and should have been as they are likely to yield better accuracy [16]. That was due to the author's lack of familiarity with SVMs at the time of this project (2005).

Acknowledgments The first author would like to acknowledge the funding received, through the REWERSE project, for his Master thesis [1] which was the basis for this paper. We would also like to thank the two anonymous referees for their comments which have improved this paper.

References

1. Santos, José C. A.: *Mining Protein Structure Data,* Master thesis. (2006)
2. Chen, H., Hu, X., Yoo, I. and Zhou, H.-X. (2004). *Classification Comparison of Prediction of Solvent Accessibility From Protein Sequences.* In Proc. Second Asia-

Pacific Bioinformatics Conference (APBC2004), Dunedin, New Zealand. CRPIT, 29. Chen, Y.-P. P., Ed. ACS. 333-338. (2004)

3. Richardson C. J., Barlow D. J. *The bottom line for prediction of residue solvent accessibility.* Protein Eng 12: 1051—1054. (1999)
4. R. Adamczak, A. Porollo and J. Meller; *Accurate Prediction of Solvent Accessibility Using Neural Networks Based Regression*, Proteins: Structure, Function and Bioinformatics, 56(4):753-67. (2004)
5. G. Gianese,. F. Bossa, and S. Pascarella. *Improvement in prediction of solvent accessibility by probability profiles*, Protein Engineering. Vol 1612, pp 987-992. (2003)
6. Correia, Marco: *Heuristic search for Protein Structure Determination*, Master thesis. (2004)
7. L. Krippahl and P. Barahona, PSICO: *Solving Protein Structures with Constraint Programming and Optimisation, Constraints, Constraints*, Vol. 7, No. 3/4, Kluwer Academic Press, pp. 317-331. (2002)
8. Chothia, C: *The nature of the accessible and buried surfaces in proteins.* J. Mol. Biol. 105, 1-14. (1976)
9. http://wwwmgs.bionet.nsc.ru/mgs/programs/crasp/texts/AA_Properties.html
10. Krippahl, L, Barahona P, *Propagating N-ary Rigid-Body Constraints* Principles and Practice of Constraint Programming, CP'2003 (Procs.), Francesca Rossi (Ed.), Lecture Notes in Computer Science, vol. 2833, Springer, pp. 452-465, October, 2003.
11. Güntert, P., Mumenthaler, C. & Wüthrich, K. *Torsion angle dynamics for NMR structure calculation with the new program DYANA.* J. Mol. Biol. 273, 283-298. (1997)
12. H.M.Berman, J.Westbrook, Z.Feng, G.Gilliland, T.N.Bhat, H.Weissig, I.N.Shindyalov, P.E.Bourne, *The Protein Data Bank*, Nucleic Acids Research, 28 pp. 235-242 (2000)
13. Murzin A. G., Brenner S. E., Hubbard T., Chothia C. (1995). *SCOP: a structural classification of proteins database for the investigation of sequences and structures*, J. Mol. Biol. 247, 536-540.
14. W. Kabsch and C. Sander. *Dictionary of Protein Secondary Structure: Pattern Recognition of Hydrogen Bonded and Geometrical Features.* Biopolymers 22: 2577-2637 (1983).
15. The CATH database: an extended protein family resource for structural and functional genomics. Pearl FM, Bennett CF, Bray JE, Harrison AP, Martin N, Shepherd A, Sillitoe I, Thornton J, Orengo CA. (2003), Nucleic Acids Research. Vol. 31, No. 1. p. 452-455.
16. Nguyen MN, Rajapakse JC. Prediction of protein relative solvent accessibility with a two-stage SVM approach. Proteins. 2005 Apr 1;59(1):30-7.