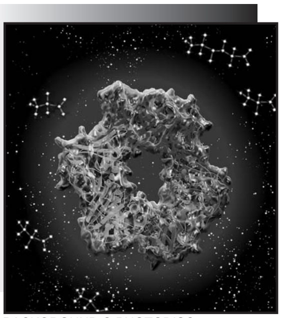


BACKGROUND © PHOTODISC, FOREGROUND IMAGE: U.S. DEPARTMENT OF ENERGY GENOMICS: GTL PROGRAM, HTTP://WWW.ORNL.GOV.HGMIS

Predicting Secondary Structures of Proteins

Recognizing Properties of Amino Acids with the Logical Analysis of Data Algorithm

BY JACEK BŁAŻEWICZ, PETER L. HAMMER, AND PIOTR ŁUKASIAK

An important assumption of all protein prediction methods is that the amino acid sequence completely and uniquely determines the three-dimensional (3-D) structure of protein. Proof that protein structure is dictated by the amino acid sequence alone is based on experiments first carried out by C. Anfinsen [2].

This assumption is supported by the following experimental evidence. If one unfolds a protein in vitro, such that no other substances are present, and then releases it, the protein immediately folds back to the same 3-D structure it had before. This folding process takes less than a second. Therefore, it seems that all the information necessary for the protein to achieve its "native structure" is contained in its amino acid sequence. The sentence above is not true for all proteins because some proteins need "auxiliary molecules" to fold.

The structural features of proteins have been divided into levels. The first level of the protein structure, called the *primary structure*, refers just to the sequence of amino acids in the protein. Polypeptide chains can sometimes fold into regular structures (i.e., structures which are the same in shape for different polypeptides) called *secondary protein structures*. The secondary structures are very simple and regular (e.g., the loop of an α-helix structure or the back and forth of a β-sheet structure). The final shape of a protein is made up of secondary structures, perhaps supersecondary structural features, and some apparently random conformations. This overall structure is referred to as the tertiary structure. Finally, many biological proteins are constructed of multiple polypeptide chains. The way these chains fit together is referred to as the quarternary structure of the protein.

Because protein secondary structure prediction was one of the first and most important problems faced by computer learning techniques, there are many methods which have been developed to solve that problem. These methods can be divided into three groups based on the information they need to predict secondary structure. Methods from the first group make predictions based on information coming from a single amino acid, either in the form of a statistical tendency to appear in an α-helix (H), β-strand (E), and coil (C) region [3] or in the form of explicit biological expert rules [4]. Methods from the second group take into account local interactions by means of an input-sliding window with encoding. Values in the output layer identify each amino acid as belonging to one of three states: α-helix, β-strand, and coil. Methods from the third group exploited the information coming from homologous sequences. This information is processed first by performing a multiple alignment between a set of similar sequences and extracting a matrix of profiles (PSSM). The first method to incorporate profile-based inputs and achieve more than 70% in accuracy was PHD [5]. The method is composed of cascading networks. Prediction accuracy can be improved by combining more than one prediction method [6], [7]. Another well-known profile-based methods is PSIPRED (protein secondary structure prediction tool based on position-specific scoring matrices) [8], which uses two neural networks to analyze profiles generated from a PSI-BLAST search, JNet [9], and SecPred. An alternative adaptive model is presented in [10]. One can find other methods that are not strictly based on neural network implementations. NNSSP (nearest neighbor secondary structure prediction) [11] uses a nearest-neighbor algorithm where the secondary structure is predicted using multiple sequence alignments and a simple jury decision method. The Web server JPred [12] integrates six different structure prediction methods and returns a consensus based on the majority rule. The program DSC (discrimination of protein secondary structure class) [13] combines several explicit parameters to get a meaningful prediction. It runs the GOR3 algorithm [3] on every sequence to provide mean potentials for the three states. The program PREDATOR [14] uses amino acid pair statistics to predict hydrogen bonds between neighboring β-strands and between amino acids in helices.

As one can see, most of the methods use homology as the important factor to determine the secondary and then the tertiary structure of a protein. Unfortunately, if a new protein sequence that has no homology with a known protein has been recognized, results obtained by these methods can include mistakes.

In this article, the Logical Analysis of Data (LAD) algorithm was applied to recognize which amino acids properties could be analyzed to deliver additional information, independent from protein homology, useful in determining the secondary structure of a protein.

 0739-5175/05/$20.00©2005IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.

There are several lines of research that point to the importance of HRV in emotion and health.

Algorithms and Methods

The structure of a protein may be represented hierarchically at four structural levels, but only the first two levels are useful for achieving the goal of the analysis described in this article.

The primary structure of a protein is the sequence of amino acids in the polypeptide chain; it can be represented as a string on the finite alphabet Σ_{aa}, with $|\Sigma_{aa}| = 20$.

Let $\Sigma_{aa} = \{A,C,D,E,F,G,H,I,K,L,M,N,P,Q,R,S,T,Y,V,W\}$ be a set of all amino acids, each letter corresponding to a different amino acid. Based on the amino acid sequence in a protein, one can create its relevant sequence of amino acids by replacing an amino acid (primary structure) in the chain with its code in the Latin alphabet. As a result, a word on the amino acid's alphabet is received.

The word s is called a protein primary structure, on the condition that letters in this word are in the same order as the amino acids in the protein chain. Let the length of the word s be denoted as $C(s)$ and let $A(s, j)$ denote an element of word s, where j is an integer number from the set $[1, C(p)]$.

The protein secondary structure refers to the set of local conformation motifs of the protein and schematizes the path followed by the backbone in the space. The secondary structure of a protein is built from three main classes of motifs: α-helix, β-strand, and loop (or coil). An α-helix is built up from one continuous region in the sequence through the formation of hydrogen bonds between amino acids in positions i and $i+4$. A β-strand does not represent an isolated structural element by itself, because it interacts with one or more β-strands (which can be distant in sequence) to form a pleated sheet called a β-*sheet*. Strands in a β-sheet are aligned adjacent to each other such that their amino acids have the same biochemical direction (parallel β-sheet) or have alternating directions (antiparallel β-sheet). Often connecting α-helices and β-strands are loop regions, which can significantly vary in length and structure, having no fixed regular shape as the other two elements. Every amino acid in the sequence belongs to one of the three structural motifs; therefore, the protein secondary structure can be reduced to a string on the alphabet $\Sigma_{ss} = \{H; E; C\}$, having the same length as the protein primary structure.

A secondary structure is represented here by a word on the relevant alphabet of secondary structures Σ_{ss}; each type of secondary structure has its own unique letter. One can denote this word by d, where the length of word d is equal to the length of word s.

Now, one may define the problem as finding a secondary structure of a protein (word d) based on the protein primary structure (i.e., word s). Moreover, for each element $A(s, j)$ one should assign an element $A(d, j)$ so that the obtained protein secondary structure r is as close as possible to a real secondary structure of the considered protein.

Several standard performance measures were used to assess the accuracy of the prediction of protein secondary structures. The measure of the three-state overall percentage of correctly predicted amino acids is usually defined by Q_3 as follows:

$$Q_3\,(\%) = \frac{\sum_{i\in\{H,E,C\}} \text{number of residues correctly predicted in state } i}{\sum_{i\in\{H,E,C\}} \text{number of residues observed in state } i} * 100. \quad (1)$$

The segment overlap measure (*SOV*) [15], [16] is calculated as shown below:

$$SOV = \frac{1}{N} \sum_{i\in\{H,E,C\}} \sum_{s(i)} \left(\frac{\text{minov}(s_1, s_2) + \delta}{\text{maxov}(s_1, s_2)} * len(s_1) \right) 100, \quad (2)$$

where $S(i)$ is the set of all overlapping pairs of segments (s_1, s_2) in conformation state i, $len(s_1)$ is the number of amino acids in segment s_1, minov(s_1, s_2) is the length of the actual overlap, and maxov(s_1, s_2) is the total extent of the segment.

The LAD method [17] has been widely applied to the analysis of a variety of real-life data sets classifying objects into two sets. It is not possible to use the original LAD method [17]–[21] directly for the considered problem. The first problem lies in the input data representation. Here, one has a sequence of amino acids, but to use the LAD approach, one should have a set of observations. Each observation must consist of a set of attributes, and all of them should be in a number format. If all of them are written as binary, one can resign from the binarization stage; however, that is not the case here, and the binarization procedure must be applied. The second problem lies in the number of classes considered in an original approach where a classification into two classes has been introduced. The proposition of an extension of the LAD into more than two classes is presented in Figure 1 [22].

Because of a complexit of the LAD algorithm [23], it is hard to present all aspects of this method. The most important ones are described below.

To make analysis more understandable, one can introduce the following terminology:

➤ *observation:* a point in a k-dimensional space $(k = 1, 2, \ldots, p)$

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.

- *database:* a set of p observations
- attribute i: each dimension of the k-space ($i = 1, 2, \ldots, k; \leq$)
- *class*: a subset of the database and as a *cut point* (x, i), value x for attribute i.

The binarization stage is needed only if data are in numerical (not binary) or nominal formats (e.g., color, shape, etc.).

The simplest way to transform a numerical attribute into a binary attribute (or attributes) is the one-cut-per-change method (3) as follows:

For two observations $\boldsymbol{a_i}$ and $\boldsymbol{b_i}$ belonging to different classes

$$a_i < x = \frac{a_i + b_i}{2} < b_i, \quad (3)$$

and there is no observation $\boldsymbol{c}$ with $a_i < c_i < b_i$.

To make such problems useful for LAD, one has to transform all data into a binary format. As a result of this stage, all attributes for each observation are changed into binary attributes. After the binarization phase, all of the observations that belonged to different classes are still different when binary attributes are taken into account.

Every pattern is defined by a set of conditions; each involves only one of the variables. For example, if pattern P_1 is defined by

$$x_{-3} > -0.705, x_{-1} > 0.285,$$
$$x_0 < 0.065, x_{+2} < -0.620$$

using values from hydrophobicity scale (pi-r), then the meaning is as follows: structure H should appear for an amino acid situated in position a_0 if, simultaneously, the value of the hydrophobicity scale is: greater than –0.705 for the amino acid situated in position a_{-3}; greater than 0.285 for the amino acid situated in position a_{-1}; smaller than 0.065 for the amino acid situated in position a_0; and smaller than –0.620 for the amino acid situated in position a_{+2} (see Table 1). The precise definition of a pattern P_1 involves two requirements. First, there should be no observations belonging to other classes that satisfy the conditions describing P_1, and, on the other hand, a huge number of observations belonging to class H should satisfy the conditions describing P_1.

Clearly, the satisfaction of the condition describing P_1 can be interpreted as a sufficient condition for an observation to belong to class H.

The observation is covered by a pattern if it satisfies all the conditions describing P_1. For the pattern-generation stage, it is important not to miss any of the "best" patterns. The pattern-generation procedure is based on the use of combinatorial enumeration techniques, which can follow a breadth first

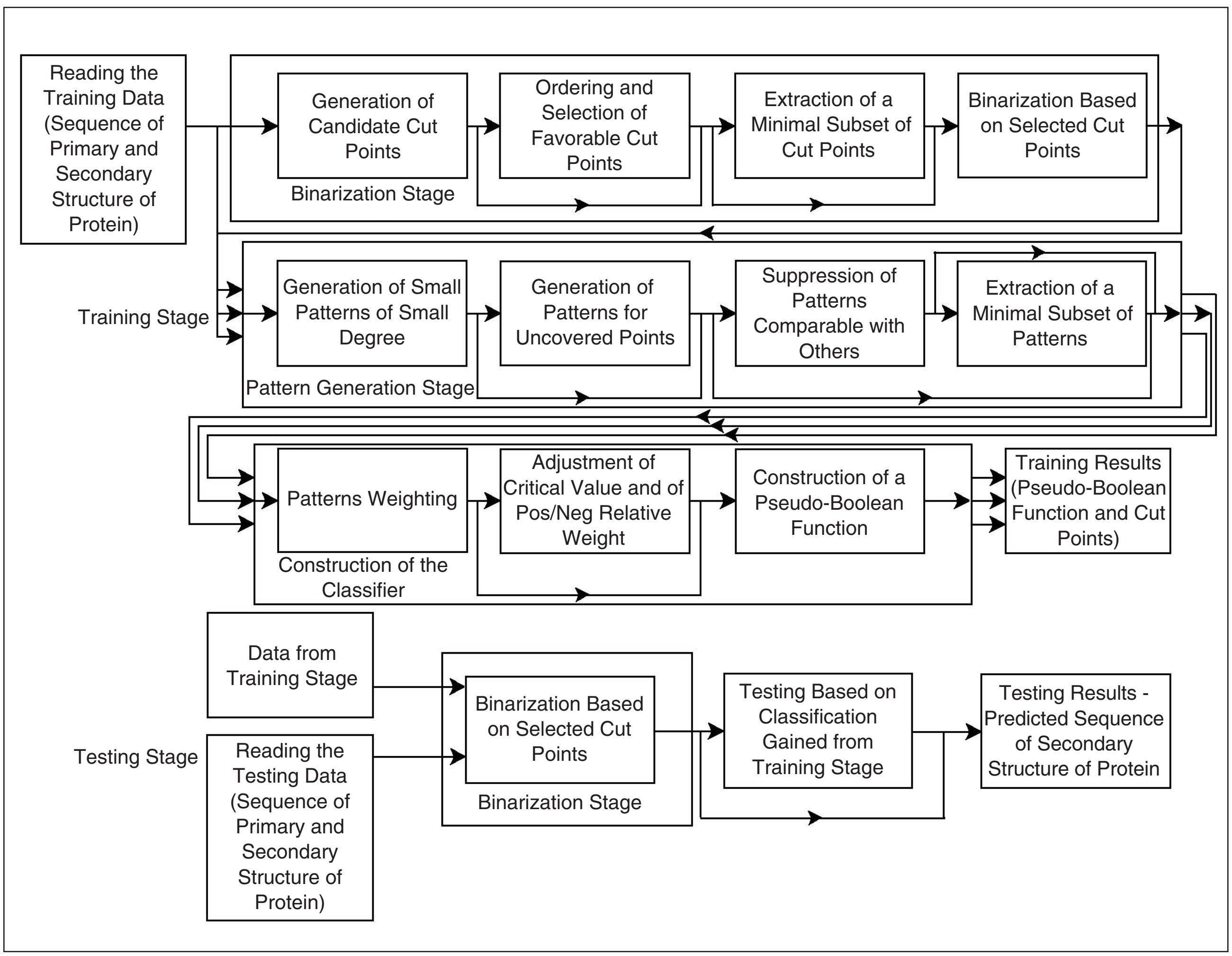


Fig. 1. The modified Logical Analysis of Data (LAD) stages.

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.

search strategy (for the patterns of up to degree 8) and depth first search strategy (for other patterns).

For any particular class there are numerous patterns which cover only observations belonging to that class. The list of these patterns is too long to be used in practice. Therefore, we restricted our attention to a subset of these patterns, called the [*class_indicator*] *model* (H model). Similarly, if one studied those observations that do not belong to the particular class, one can consider the not-H model.

An H model is simply a list of patterns associated with the observations that belong only to class H, having the following two properties:

➤ if an observation is covered by at least one of the patterns from the the H model in position a_0, class H appears for that observation
➤ if an observation is covered by none of the patterns from the H model in position a_0, class H does not appear for that observation.

Before this stage is performed, every positive (or negative) observation point is covered by at least one positive (or negative) pattern, and it is not covered by any negative (or positive) patterns that have been generated. Therefore, it can be expected that an adequately chosen collection of patterns can be used for the construction of a general classification rule. This rule is an extension of a partially defined Boolean function, and will be called a *theory* below.

A good classification rule should capture all the significant aspects of the phenomenon.

The simplest method of building a theory consists in defining a weighted sum (4) of positive and negative patterns, and classifying new observations according to the value of the following weighted sum:

$$\Delta = \sum_{k=1}^{r} \omega_k^+ P_k + \sum_{l=1}^{s} \omega_l^- N_l, \qquad (4)$$

where

➤ ω_k^+ is a nonnegative weight for positive pattern P_k (for $1 \leq k \leq r$), r is a number of positive patterns
➤ ω_l^- is a nonpositive weight for negative pattern N_l (for $1 \leq l \leq s$), s is a number of negative patterns.

See [24], [17] for a more detailed description of the LAD method.

As in previous experiments (i.e., [22], [25]), at the beginning three binary one-versus-rest classifiers were constructed; here, *one* means positive class (e.g. H) and *rest* means negative class (in that case E, C), denoted as: H/~H, E/~E, C/~C. The one-versus-rest classifiers often need to deal with two

Table 1. An example of rules (a horizontal line in a cell means that the value of the attribute is not important for making a decision for that pattern).

#	a_{-3}	a_{-2}	a_{-1}	a_0	a_{+1}	a_{+2}	a_{+3}	Property
1	> 0.705	—	>0.285	<0.065	—	< 0.620	—	Hydrophobicity
2	< 0.620	< 0.130	—	>1.795	—	> 0.020	—	scale (pi-r)
3	—	>1.745	<0.195	>1.225	>1.795	—	>0.195	class H
1	—	—	<11.705	>14.195	<11.365	>14.765	<11.295	Avg. surround.
2	>15.285	—	>12.700	>12.295	—	<11.705	—	hydrophobicity
3	>15.690	<11.395	—	<13.195	>15.285	—	—	class E
1	>10.45	—	—	>9.10	<5.60	>10.45	>9.10	Polarity (p)
2	>11.95	>10.90	—	<10.45	>12.65	—	—	class C
3	—	>9.80	>8.80	>11.95	—	<6.60	—	

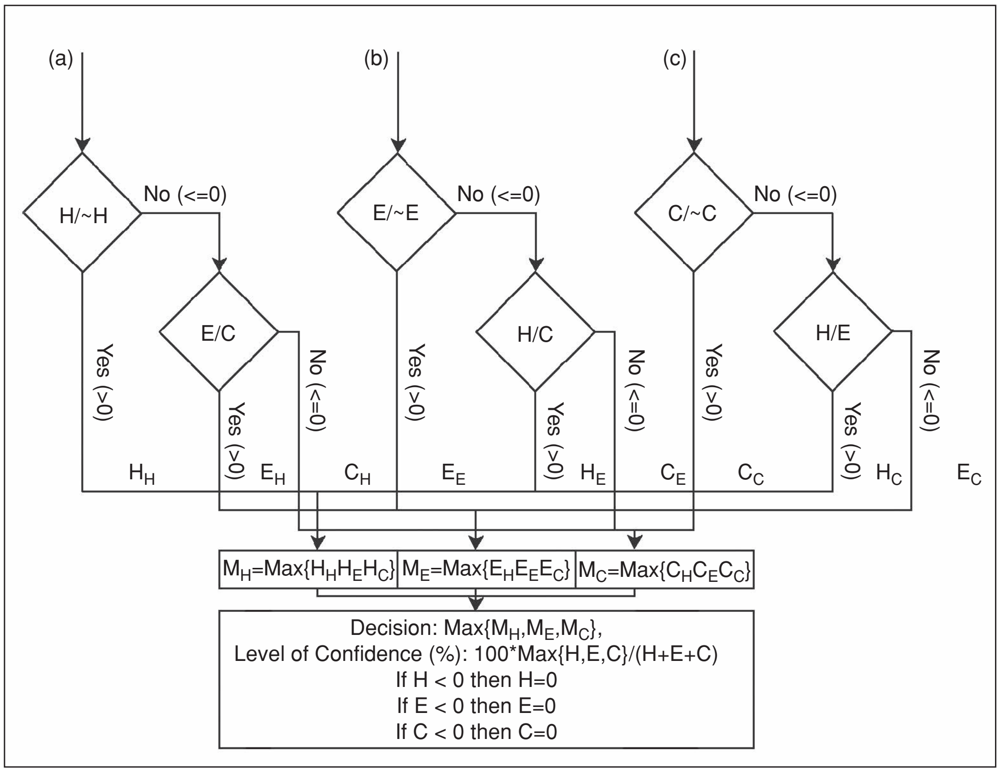


Fig. 2. The decision graphs for classifiers; each of them is made up of two binary classifiers: a) classifiers H/~H, E/C; b) classifiers E/~E, H/C; and c) classifiers C/~C, H/E.

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.

Table 2. The accuracy of the prediction of secondary structures for three classes using MODLEM (33) and LAD (ninefold cross-validation test: 9 proteins, 2,100 amino acids).

Property	Accuracy of Prediction (%)	
	MODLEM	LAD
Normalized consensus hydrophobicity scale	59.86	62.86
Mobilities of amino acids on chromatography paper (RF)	60.71	65.86
Hydrophobicity scale based on free energy of transfer (kcal/mol)	58.24	65.19
Hydrophobicity indices at pH 7.5 determined by HPLC	59.95	64.33
Average surrounding hydrophobicity	59.24	64.71
Hydrophobicity indices at pH 3.4 determined by HPLC	62.71	67.67
Retention coefficient in TFA	58.67	64.38
Hydration potential (kcal/mol) at 25°C	58.95	71.60
Retention coefficient in HPLC, pH 7.4	59.38	65.86

HPLC = high power liquid chromatography

Table 3. A set of the best properties for class H.

#	Description
1	Molecular weight of each amino acid
2	Hydrophobicity scale (pi-r)
3	Hydrophobicity scale (contact energy derived from 3-D data)
4	Hydrophilicity
5	Normalized consensus hydrophobicity scale

Table 4. A set of the best properties for class E.

#	Description
1	Average surrounding hydrophobicity
2	Bulkiness
3	Hydrophilicity scale derived from HPLC peptide retention times
4	Hydrophobicity scale (contact energy derived from 3-D data)
5	Hydrophobicity scale (pi-r)

Table 5. A set of the best properties for class C.

#	Description
1	Polarity (p)
2	Hydropathicity
3	Retention coefficient in TFA
4	Retention coefficient in HFBA
5	Hydrophobic constants derived from HPLC peptide retention times

data sets with different sizes, i.e., unbalanced training data [26]. Therefore, during experiments, three additional classifiers were added: H/E, H/C, E/C. The set of all six classifiers allows one to distinguish the observation between each of two states. However, a potential problem of the one-versus-one classifier is that the voting scheme might suffer from incompetent classifiers. One can reduce that problem by using a decision graph [27] with some modifications (as shown in Figure 2).

The protein secondary structure is assigned from the experimentally determined tertiary structure by DSSP [28], STRIDE [29], or DEFINE [30]. To implement the methods and extract the basic properties of proteins, examples were obtained from the *Dictionary of Protein Secondary Structures.*

There are many ways to divide protein secondary structures into classes. Here, we used the most popular based on information obtained from DSSP.

Data gained from the DSSP set consist of eight types of protein secondary structures: α-helix (structure denoted by H in DSSP), 3_{10}-helix (G), π-helix (I), β-strand (E), isolated β-bridge (B), turn (T), bend (S), and rest (−). The following sets of secondary structures have been created:

- helix (H) consisting of: α-helix (structure denoted by H in DSSP), 3_{10}-helix (G) and π-helix (I)
- β-strand (E) consisting of E structure in DSSP
- the rest (C) consisting of structures belonging neither to set H nor to set E.

In making a transformation from a protein sequence to the set of observations, one must assume that the main influence on the secondary structure is having amino acids situated in the neighborhood of the observed amino acid. We also took into account that some *n*-mers are known to occur always in the same structure in many proteins, while others do not. Certain 4-mers and 5-mers are known to have different secondary structures in different proteins. To fulfill this assumption and avoid naive mistakes, a concept of windows [31] was used to create a set of observations. It should be done carefully because if the size of window is too short, it may lose some important classification information and prediction accuracy; if a window is too long, it may suffer from the inclusion of unnecessary noise. For the experiments, the window of size 7 [22] was used. An example is presented here, illustrating the way a protein chain is changed into a set of observations.

Let us consider a protein chain called *4gr1* (in PDB). The first and the last 15 amino acids in the sequence are shown here:

VASYDYLVIGGGSGG . . . VAIHPTSSEELVTLR

For every amino acid the corresponding protein secondary structure in DSSP is given as follows:

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.

_EE_SEEEE__SHHH . . . **___SS_SGGGGGS__**

One may change this structure into a protein secondary structures involving three main secondary structures only in the manner depicted here:

XEEXXEEEEXXXHHH . . . **XXXXXXXHHHHHXXX**

A window of length 7 generates an observation with 7 attributes (a_{-3}, a_{-2}, a_{-1}, a_0, a_{+1}, a_{+2}, a_{+3}) representing a protein secondary structure corresponding to the amino acid located in place a_0. Of course, at this moment all values of attributes are symbols of amino acids. Secondary structures of proteins on the boundaries (the first three and the last three amino acids) have been omitted and treated as unknown observations. For example, the first observation can be constructed by amino acids **VASYDYL** and that observation describes the class for an amino acid situated in the middle (amino acid Y) – class X; the next observation is created by a window shifted one position to the right, etc.

The last step of the preprocessing is to replace in each observation symbols of amino acids (treated as attributes) with numbers representing relevant properties of amino acids. During the experiment only the physical and chemical properties of the amino acids have been taken into account. Originally, 54 properties were considered, but after a discussion with domain experts, 28 were chosen for the experiment. The chosen set seems to consist of the most important properties from a biology viewpoint. At the end of transformation, a chain consisting of *n* amino acids is transformed into a set consisting of *n*-6 observations.

Results and Discussion

During experiments to develop and test the algorithms, 20 proteins from the nonhomologous data set proposed by [1] were applied. This set consists of 126 nonhomologous proteins which can be obtained from ftp.cmbi.kun.nl/pub/molbio/data/dssp. The physico-chemical properties of amino acids were used as attributes.

Prediction accuracy for structure H was between 18–57%, and the best result was obtained using as attributes the values of molecular weight of each amino acid.

For structure E, results varied between 7–74%; the best result was acheived when average surrounding hydrophobicity was treated as an attribute of observation.

The average accuracy for structure C was between 15–69% and the best property for class C was polarity (p). The best average accuracy for all three classes was achieved using optimized matching hydrophobicity (OMH). Unfortunately it was not possible to find a single property which could serve as a universal property for detecting all secondary structure types in a protein shape, but one should not expect results like that. It would be all too easy if only one property could be responsible for the protein 3-D structure.

It seems that the accuracy of a prediction of a secondary structure for each class can be higher if a few properties with the best ability of prediction can be treated simultaneously as attributes. The average accuracy of the prediction of secondary structures of proteins can be higher if the best properties from different classes of secondary structures would have been taken into consideration simultaneously.

A comparison of the results between two different machine-learning methods (Table 2) shows that the results obtained by LAD can be treated as representative results obtained by machine-learning methods, and the properties presented in Tables 3 through 5 should be analyzed during the process of the artificial construction of a protein's 3-D shape before a homology stage.

Table 6. A comparison of different methods. Results for PHD, DSC, PREDATOR, and CONSENSUS were obtained from (7). The PHD results were obtained from (1), (15). The LAD results are from a new method proposed by authors (tested on 20 randomly selected proteins from the RS 126 benchmark data set).

Method	Q (%)	SOV
PHD	70.8	73.5
PHD	73.5	73.5
DSC	71.1	71.6
PREDATOR	70.3	69.9
NNSP	72.7	70.6
CONSENSUS	74.8	74.5
LAD	70.6	70.3

The system constructed, using LAD as its engine, generates results comparable to the best methods currently used for protein secondary structure prediction. Table 6 shows that results obtained using LAD are worse than results obtained by PHD and CONSENSUS. However, the advantage of LAD is that LAD is not a "black box" as PHD is. Rules generated by LAD can deliver important information for the understanding of the mechanism causing the phenomenon. An example of rules generated by LAD is shown in Table 1.

Conclusions

This article presents the application of a new machine-learning algorithm for the prediction of secondary structures of proteins. The results obtained from the experiments show that this method can be successfully applied. Although it is not possible to predict all the secondary structures for every protein chain (the protein backbone often folds back on itself in forming a structure, so flexibility is an important attribute that has not been taken into account during experiments), it has been shown that information included in some types of amino acid properties (presented in Tables 3–5) is important and can serve as basic information about the protein shape. Based on the experiments and protein chains taken for analysis, it can be said that the most important property for class H is the molecular weight of each amino acid, for class E it is the average surrounding hydrophobicity, and for class C it is polarity (p).

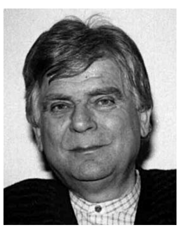
To get better results, LAD should be used as a first stage of analysis in combination with another method that is able to take into account a more detailed understanding of the physical chemistry of proteins and amino acids. It seems to be valuable and important for the prediction of protein secondary structures to construct a library of rules, which can describe the core of the considered phenomenon.

Acknowledgments

The work is supported by the State Committee for Scientific Research grant.

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.

Jacek Błażewicz received his M.Sc. in control engineering in 1974 and his Ph.D. and Dr. habil. in computer science in 1977 and 1980, respectively. He is a professor of computer science at the Poznan University of Technology and is a deputy director of the Institute of Computing Science. His research interests include algorithm design and complexity analysis of algorithms, especially in bioinformatics, as well as in scheduling theory. He has published widely in many outstanding journals, including the *Journal of Computational Biology*, *Bioinformatics*, *Computer Applications in Biosciences*, and *IEEE Transactions on Computers*. He is also the author and coauthor of fourteen monographs. Blazewicz is also an editor of the *International Series of Handbooks in Information Systems* as well as a member of the editorial boards of several scientific journals. In 1991 he was awarded EURO Gold Medal for his scientific achievements in the area of operations research. In 2002 he was elected as a corresponding member of the Polish Academy of Sciences.



Peter L. Hammer is the professor and director of RUTCOR, the Rutgers Center for Operations Research. His research interests include Boolean methods in operations research and related areas, Boolean and pseudo-Boolean functions, discrete optimization, theory of graphs and networks, logical design, switching theory, and threshold logic. He has published widely in many outstanding journals for the fields of operations research. He is also an editor of many well-known journals for combinatorial optimization.

Piotr Łukasiak is an assistant professor at the Poznan University of Technology, Institute of Computing Science and in Institute of Bioorganic Chemistry, Laboratory of Bioinformatics Polish academy of Sciences, Ph.D. in Computer Science (Poznan University of Technology). He was also a holder of fellowship from Max-Planck Institute, Germany. His research interests include algorithm design and complexity analysis of algorithms, computational biology, combinatorial optimization, and machine learning.

Address for Correspondence: Piotr Łukasiak, Institute of Computing Sciences, Poznan University of Technology, Piotrowo 3A, 60-965 Poznan, Poland. Phone: +48 61 8528503 ext. 285. Fax: +48 61 8771525. E-mail: Piotr.Lukasiak@cs.put.poznan.pl.

References

[1] B. Rost, C. Sander, "Prediction of protein secondary structure at better than 70% accuracy," *J. Mol. Biol.*, vol. 232, pp. 584–599, 1993.

[2] C.B. Anfinsen, "Principles that govern the folding of protein chains," *Science*, vol. 181, pp. 223–230, 1973.

[3] J. Garnier, D. Osguthorpe, and B. Robson, "Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins," *J. Mol. Biol.*, vol. 120, pp. 97–120, 1978.

[4] V. Lim, "Algorithms for prediction of α-helical and β-structural regions in globular proteins," *J. Mol. Biol.*, vol. 88, pp. 873–894, 1974.

[5] B. Rost, "PHD: Predicting one-dimensional protein structure by profile based neural networks," *Meth. Enzymol.*, vol. 266, pp. 525–539, 1996.

[6] X. Zhang, J.P. Mesirov, and D.L. Waltz, "Hybrid system for protein secondary structure prediction," *J. Mol. Biol.*, vol. 225, pp. 1049–1063, 1992.

[7] J.A. Cuff and G.J. Barton, "Evaluation and improvement of multiple sequence methods for protein secondary structure prediction," *Proteins: Struct. Funct. Genet.*, vol. 34, pp. 508–519, 1999.

[8] D. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *J. Mol. Biol.*, vol. 292, pp. 195–202, 1999.

[9] J.A. Cuff and G.J. Barton, "Application of multiple sequence alignment profiles to improve protein secondary structure prediction," *Proteins*, vol. 40, pp. 502–511, 2000.

[10] P. Baldi, S. Brunak, P. Frasconi, G. Soda, and G. Pollastri, "Exploiting the past and the future in protein secondary structure prediction," *Bioinformatics*, vol. 15, pp. 937–946, 1999.

[11] A. Salamov and V. Solovyev, "Prediction of protein secondary structure by combining nearest-neighbor algorithms and multiple sequence alignment," *J. Mol. Biol.*, vol. 247, pp. 11–15, 1995.

[12] J.A. Cuff, M.E. Clamp, A.S. Siddiqui, M. Finlay, and G.J. Barton, "Jpred: A consensus secondary structure prediction server," *Bioinformatics*, vol. 14, pp. 892–893, 1998.

[13] R. King and M. Sternberg, "Identification and application of the concepts important for accurate and reliable protein secondary structure prediction," *Prot. Sci.*, vol. 5, pp. 2298–2310, 1996.

[14] D. Frishman and P. Argos, "Seventy-five percent accuracy in protein secondary structure prediction" *Proteins*, vol. 27, pp. 329–335, 1997.

[15] B. Rost, C. Sander, and R. Schneider, "Redefining the goals of protein secondary structure prediction," *J. Mol. Biol.*, vol. 235, pp. 13–26, 1994.

[16] A. Zemla, C. Venclovas, K. Fidelis, and B. Rost, "A modified definition of SOV, a segment based measure for protein secondary structure prediction assessment," *Proteins: Struct. Funct. Genet.*, vol. 34, pp. 220–223, 1999.

[17] P.L. Hammer, "Partially defined boolean functions and cause-effect relationships," presented at the International Conference on Multi-Attribute Decision Making Via OR-Based Expert Systems, Passau, Germany, 1986.

[18] E. Boros, P.L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, and I. Muchnik, "An implementation of logical analysis of data," *Rutcor Res. Rep.*, Rep. 22–96, 1996**.**

[19] E. Boros, P.L. Hammer, A. Kogan, E. Mayoraz, and I. Muchnik, "Logical analysis of data—overview," *Rutcor Res. Rep.*, Rep. 1–94, 1994.

[20] Y. Crama, P.L. Hammer, and T. Ibaraki, "Cause-effect relationships and partally defined Boolean functions," *Ann. Oper. Res.*, vol. 16, pp. 299–326, 1998.

[21] E. Mayoraz, "C++ tools for logical analysis of data," *Rutcor Research Report*, Rep. 1–95, 1995.

[22] J. Blazewicz, P.L. Hammer, and P. Lukasiak, "Logical Analysis of Data as a predictor of protein secondary structures," in *Bioinformatics of Genome Regulations and Structure*, N. Kolchanov and R. Hofestaedt, Eds. Norwell, MA: Kluwer, 2004, pp. 145–154.

[23] O. Ekin, P.L. Hammer, and A. Kogan, "Convexity and logical analysis of data," *Rutcor Research Report*, Rep. 5–98, 1998.

[24] E. Boros, P.L. Hammer, T. Ibaraki, and A. Kogan, "Logical analysis of numerical data," *Rutcor Research Report*, Rep. 4–97, 1997.

[25] J. Blazewicz, P.L. Hammer, and P. Lukasiak, "Prediction of protein secondary structure using Logical Analysis of Data algorithm," *Comput. Methods Sci. Technol.*, vol. 7, no. 1, pp. 7–25, 2001.

[26] C.W. Hsu and C.J. Lin, "A comparison of methods for multi-class support vector machines," *IEEE Trans. Neural Networks*, vol. 13, pp. 415–425, 2002.

[27] S.J. Hua and Z.R. Sun, "A novel method of protein secondary structure prediction with high overlap measure: Support vector machine approach," *J. Mol. Biol.*, vol. 308, pp. 397–407, 2001.

[28] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: Pattern recognition of hydrogen bonded and geometrical features," *Biopolymers*, vol. 22, pp. 2577–2637, 1983.

[29] D. Frishman and P. Argos, "Knowledge based secondary structure assignment," *Proteins: Struct. Funct. Genet.*, vol. 23, pp. 566–579, 1995.

[30] F.M. Richards and C.E. Kundrot, "Identification of structural motifs from protein coordinate data: Secondary structure and first level supersecondary structure" *Proteins: Struct. Funct. Genet.*, vol. 3, pp. 71–84, 1988.

[31] R.D. King and M.J.E. Sternberg, "Machine learning approach for the prediction of protein secondary structure," *J. Mol. Biol.*, vol. 216, pp. 441–457, 1990.

[32] J.W. Grzymala-Busse, "LERS—A system for learning from examples based on rough sets," in *Intelligent Decision Support—Handbook of Applications and Advances of the Rough Set Theory*, R. Slowinski, Ed. Norwell, MA: Kluwer, 1992, pp. 3–18.

[33] J.W. Grzymala-Busse and J. Stefanowski, "Three approaches to numerical attribute discretization for rule induction," *Int. J. Intell. Syst.*, vol. 16, pp. 29–38, 2001.

Authorized licensed use limited to: IEEE Xplore. Downloaded on April 15, 2009 at 07:00 from IEEE Xplore. Restrictions apply.