Protein Classification Using Neural Networks

Edgardo A. Ferrán and **Pascual Ferrara**
Sanofi Elf Bio Recherches
Labège Innopole, BP 137
31676 Labège, France

Bernard Pflugfelder
Societé Nationale Elf Aquitaine
26, avenue des Lilas
64008 Pau, France

Abstract

We have recently described a method based on Artificial Neural Networks to cluster protein sequences into families. The network was trained with Kohonen's unsupervised–learning algorithm using, as inputs, matrix patterns derived from the bipeptide composition of the proteins. We show here the application of that method to classify 1758 protein sequences, using as inputs a limited number of principal components of the bipeptidic matrices. As a result of training, the network self–organized the activation of its neurons into a topologically ordered map, in which proteins belonging to a known family (immunoglobulins, actins, interferons, myosins, HLA histocompatibility antigens, hemoglobins, etc.) were usually associated with the same neuron or with neighboring ones. Once the topological map has been obtained, the classification of new sequences is very fast.

Introduction

The continuous increase in the number of known nucleic acid and protein sequences during the last years has led to the development of advanced computational tools to search for sequence similarities in macromolecular databases. There are powerful algorithms for comparing two (Needleman & Wunsch 1970) or more sequences (Gribskov, McLachlan & Eisenberg 1987, Corpet 1988). In general, these comparisons involve sequence alignments, allowing for the existance of gaps in each sequence. Although these methods are sensitive, they are extremely time consuming. Faster but less sensitive algorithms to identify related proteins have also been proposed (Lipman & Pearson 1985, Altschul & Lipman 1990, Altschul et al. 1990). In spite of these developments, the search for sequence similarities in macromolecular databases is still a subject of major concern, because sequencing data keep increasing at high speed as a consequence of many genome sequencing projects (Watson 1990; Sulston et al. 1992, Maddox 1992) and searching time, in standard algorithms, is usually proportional to database size. One possible strategy to attack this problem is to cluster macromolecular sequences into families and then compare new sequences only with consensus patterns representing each family. Thus, as the number of macromolecular families is expected to growth slower than the number of sequences, searching time should not be so dependent on database size. Recently, two different neural–network based methods following this approach has been proposed (Ferrán & Ferrara 1991a, Wu et al. 1992).

Artificial Neural Networks (ANNs) are simplified models of the nervous system, in which neurons are considered as simple processing units linked with weighted connections called synaptic efficacies. These weights are gradually adjusted according to a *learning* algorithm.

During recent years, ANNs have been applied as a computational tool to a large number of different fields. In most cases, a feed–forward architecture of the network is used to predict the relationship between inputs and outputs, after "learning" some known examples. The corresponding final set of synaptic connections is determined using a *supervised* learning algorithm: usually, the *delta rule* algorithm (Rosenblatt 1962) for networks having only one layer of adaptable synaptic efficacies and the *backpropagation* algorithm (Le Cun 1985, Rumelhart, Hinton & Williams 1986) for multi–layered networks. In particular, feed–forward ANNs have been applied to the analysis of biological sequences (von Heijne 1991, Petersen et al. 1990, Hirst & Sternberg 1992) by considering some representation of the sequence as input to the network. For protein sequences, this approach has been used to predict immunoglobulin domains (Bengio & Pouliot 1990), surface exposure of amino acids (Holbrook, Muskal & Kim 1990), disulfide–bonding states of cysteines (Muskal, Holbrook & Kim 1990), signal peptides (Ladunga et al. 1991), ATP–binding motifs (Hirst & Sternberg 1991), water binding sites (Wade, Bohr & Wolynes 1992), three–dimensional (Bohr et al. 1990) and secondary structures of proteins (Qian & Sejnovski 1988, Bohr et al. 1988, Holley & Karplus 1989, McGregor, Flores & Sternberg 1989, Andreassen

et al. 1990, Kneller, Cohen & Langridge 1990, Vieth & Kolinski 1991, Stolorz, Lapedes & Xia 1992, Muskal & Kim 1992, Zhang, Mesirov & Waltz 1992, Hayward & Collins 1992, Vieth et al. 1992, Rost & Sander 1992, Sasagawa & Tajima 1993) and to recognize distantly related protein sequences (Frishman & Argos 1992). For nucleic acid sequences, it has been used to predict DNA–binding sites or promoters (Stormo et al. 1982, Lukashin et al. 1989, O'Neill 1991, 1992, Demeler & Zhou 1991, Horton & Kanehisa 1992), mRNA splice sites (Brunak, Engelbrecht & Knudsen 1990, 1991, Engelbrecht, Knudsen & Brunak 1992) and coding regions in DNA (Lapedes et al. 1990, Uberbacher & Mural 1991, Farber, Lapedes & Sirotkin 1992, Snyder & Stormo 1993).

In a very different approach, as the one proposed by Kohonen (1982), the neural network self–organizes its activation states into topologically ordered maps, using an *unsupervised* learning method. These maps result from an information compression that only retains the most relevant common features of the set of input signals. This approach has been applied to detect signal peptide coding regions (Arrigo et al. 1991) and to cluster molecules of analogue structure into families of similar activity (Rose, Croall & MacFie 1991). We have proposed (Ferrán & Ferrara 1991a,b 1992a) a method based on Kohonen's algorithm to cluster protein sequences into families according to their degree of sequence similarity. The network was trained using, as inputs, matrix patterns of 20x20 components derived from the bipeptide composition of the protein sequences. This naive representation of the whole sequence information has also been successfully applied to classify proteins with statistical techniques (Nakayama, Shigezumi & Yoshida 1988, Van Heel 1991) and allowed us to feed the network with a constant number of inputs, regardless the protein length. We have tested the method by considering both small ($\approx$ 10 sequences) and large ($\approx$ 450 sequences) learning sets of well–defined protein families (Ferrán & Ferrara 1991a, 1992a). For small learning sets, we have also shown that the trained network is able to correctly classify mutated or incomplete sequences of the learned proteins (Ferrán & Ferrara 1991a). We have also found, using a learning set of 76 cytochrome *c* sequences belonging to different species, that the time evolution of the map during learning roughly resembles the phylogenetical classification of the involved species (Ferrán & Ferrara 1992b). We have also described a large–scale application of the method, in which a network composed with 225 neurons was trained to classify a learning set of 1758 bipeptide matrices (Ferrán, Pflugfelder & Ferrara 1992; Ferrán & Ferrara 1993). This set corresponded to all the human protein sequences included in the SwissProt database (release 19.0, 8/91), whose lengths were greater than 50 amino acids. When learning proceeded during 500 learning cycles or *epochs*, it took about 100 CPU–hours on a SUN 4/360 computer (16 MIPS, 2.6 MFLOPS) to form the topological map for the set of 1758 human proteins. A faster learning protocol, involving only 30 epochs (6.7 SUN CPU–hours), also provided a suitable classification (Ferrán & Ferrara 1992c, 1993). In both cases, known families of proteins were usually found to be associated either with the same neuron or with neighboring ones, on the final topological map. Although network training is time consuming, once the topological map is obtained the classification of a new protein is very fast (Ferrán & Ferrara 1992c). Other matrix protein representations, taking into account the physico–chemical properties of amino acids, have also been explored, to reduce the computing time required for the learning procedure (Ferrán & Ferrara 1993).

We have also compared the classification obtained by the above described ANN approach with the one that results from a statistically based clustering method (Ferrán & Pflugfelder 1993). The statistical method consisted of three stages (Pflugfelder & Ferrán 1992): i) principal component analysis of the set of bipeptidic matrices, ii) determination of the optimal number M of clusters, using only a limited number of principal components, and iii) final classification of the protein sequences into M clusters. We have shown that the results from the statistical method can not only be used to validate the results obtained with the ANN approach but also to reduce the number of inputs of the network and to choose, in a more reasonable way, the number of neurons (Ferrán & Pflugfelder 1993). In general, the hybridation of the statistical and ANN approaches can decrease the computing time required to train the network. We show here a large–scale application of the method, in which the same learning set of 1758 human protein sequences mentioned above is classified using a limited number of principal components of the bipeptidic matrices as protein representation.

Methods

Bipeptide Representation

In this section we summarize the standard formalism of the method we have previously proposed [see (Ferrán & Ferrara 1991a, 1992a) for a detailed description].

In general, we consider a two–dimensional network, that is, one layer of $N_x N_y$ neurons. Each neuron receives, as input signals, a pattern of 20 x 20 components ξ_{kl}, obtained from the bipeptide composition of the protein to be learned. The 400 values of the corresponding synaptic efficacies that weight the input signals are the components of a synaptic vector associated with each neuron. We denote by $\vec{m}_{ij}$ the synaptic vector of the neuron positioned in the (i, j) site of the output layer. In what follows, we will identify each neuron directly by its position. At the beginning, all synaptic vector components $\mu_{ij,kl}$ are real numbers randomly taken from the interval [0,1]. Both, input patterns and synaptic vectors, are normalized to unitary vectors. Each protein pattern is presented as input to

the network and the neuron having the closest synaptic vector to the protein pattern (the *winner* neuron) is selected. Then, the synaptic vectors of all neurons belonging to a winner neighborhood N_w are changed in order to bring these vectors closer to the vector of input signals:

$$\mu_{ij,kl}(t+1) = \mu_{ij,kl}(t) + \alpha(t)[\xi_{kl}(t) - \mu_{ij,kl}(t)], \quad \forall \text{ neuron } (i,j) \in N_w,$$

where $0 < \alpha(t) < 1$. All protein patterns of the learning set are repeatedly processed by the network, in the same sequential order. Each processing cycle of the whole learning set is called an epoch. As learning proceeds, α is linearly or exponentially decreased every Δt_α epochs ($0 < a < 1$):

$$\alpha(t + \Delta t_\alpha) = \alpha(t) - a,$$
$$\alpha(t + \Delta t_\alpha) = a\alpha(t),$$

and the winner neighborhood is shrunk, from the whole network to the winner neuron, every Δt_v epochs.

Once learning has been accomplished, each sequence of the learning set is finally associated with the neuron having the closest synaptic vector. Thus, each synaptic vector of the trained network may be considered as a "consensus pattern" for the set of bipeptide matrices of protein sequences associated with the corresponding neuron.

Principal Components Representation

Each of the K bipeptidic matrices of the learning set can be considered as a single point $\vec{x}_k$ in a 400–dimensional space ($k = 1, 2, ..., K$). Since the above described selection of the winner neuron involves the computation of the distance between $\vec{x}_k$ and each synaptic vector $\vec{m}_{ij}$, the computing time required to train the network depends on the dimensionality of the vector space. To reduce this time we have pre–processed the inputs of the network, performing a Principal Components Analysis (PCA). This standard multivariate technique, originated by Pearson (1901) and later developed by Hotelling (1933), is usually applied to uncorrelate and to reduce the number of variables describing a set of individuals (Rao 1960, 1964). It consists in finding the orientations of the *principal axes* in which the covariance matrix of that set is diagonal, when the origin of coordinates is placed in the center of gravity $\vec{x} = \sum_{k=1,K} \vec{x}_k / K$ of the set of individuals. These orthogonal orientations are given by the eigenvectors of the covariance matrix, which can be obtained by a rotation of the initial axes. Therefore, each individual can be described by a vector whose 400 components F_l ($l = 1, 2, ..., 400$) are its orthogonal projections along the principal axes. In addition, these axes can be sorted in a hierarchical way, according to their contributions to the inertia of the set of individuals, which are given by the corresponding eigenvalues λ_l. This hierarchical organization may be used to take into account only a reduced set of axes. This reduced set should permit to describe the set of bipeptidic matrices without much loss of information.

A usual criterion to determine how many principal components must be considered is to take into account those corresponding to a given amount of the accumulated inertia (for instance, 80% or 90%). Other standard criterion is to consider all the principal components having eigenvalues greater than one. In the present paper we follow the last criterion. Thus, we use as inputs to the network the unitary vectors whose components are the first n principal components of the bipeptidic matrices ($\lambda_l \geq 1, \forall l \leq n$). The rest of the learning algorithm is the same as before.

Results

We have performed a Principal Component Analysis on a learning set of 1758 bipeptidic matrices, using the PRINCOMP procedure of the SAS package (SAS Institute 1985, Chapter 28). This set of matrices corresponded to all the human protein sequences stored in the SwissProt database (release 19.0, 8/91), whose lengths were greater than 50 amino acids. Table 1 shows the accumulated inertia for the n principal components. The eigenvalues of the first 59 principal components (corresponding to $\approx 70\%$ of the total inertia) are greater than one. As it can be seen, only a small number of principal components is needed to explain a large amount of the inertia of the whole cloud of 1758 individuals.

Then, we trained a two–dimensional network of 15 x 15 neurons, using as inputs the first 60 principal components of the bipeptidic matrices. In Fig.1 we indicate the number of sequences having each neuron as winner and the position of a few known families of proteins on the resulting topological map. The training stage of the network, with a fast protocol (30 epochs), took only about 50 SUN CPU–minutes. Therefore, computing time is reduced about 8 times, when the number of inputs is diminished from 400 to 60. As a consequence of Kohonen's learning algorithm, similarity relationships between protein sequences have been mapped into neighborhood relationships of neural activity on the two–dimensional layer of neurons. For example, almost all immunoglobulins were placed in the left side of the map. Inside this zone, they were subclassified according to the type of immunoglobulin chain: heavy chains were placed in neurons (12,1), (11,1), (11,2) and (10,1); λ–chains in neurons (9,1), (9,2), (8,1), (8,2), (7,1), (7,2) and (5,2); κ–chains in neurons (7,1), (7,2), (6,1), (6,2) and (5,1). Note that there is an overlapping of the λ– and κ–chain subfamilies in neurons (7,1) and (7,2). The sequences corresponding to the constant immunoglobulin regions were associated with neurons both inside [(8,1) and (8,3)] and outside [(1,14), (2,1), (3,5), (3,13), (3,15), (5,13) and (7,14)] of this "immunoglobulin zone". When the bipeptidic matrices were considered as inputs, these

n	λ_n	Acc. Inertia (%)
1	129.73	32.4
2	24.35	38.5
3	12.26	41.6
4	9.79	44.0
5	7.17	45.8
6	6.40	47.4
7	5.15	48.7
8	4.24	49.8
9	3.62	50.7
10	3.46	51.4
20	1.96	57.4
30	1.51	61.6
40	1.26	65.0
50	1.12	67.9
60	0.99	70.6
70	0.91	73.0
80	0.83	75.1
90	0.76	77.1
100	0.71	79.0
106	0.68	80.0
184	0.38	90.0

Table 1: Principal Component Analysis. Eigenvalues λ_n and percentages of accumulated inertia corresponding to the first n principal components of the 1758 human bipeptidic matrices.

sequences were placed in the boundary of that zone (Ferrán & Ferrara 1993). Similarly, most of the zinc–finger proteins were placed in the upper left corner of the map [neurons (1,3), (1,4), (1,5), (2,3), (2,4), (2,5) and (3,5)]. The remaining five were placed in neuron (12,15). This subclassification has not been found when the proteins were represented by their bipeptidic matrices. All interferon–α precursors were placed in only one neuron [neuron (6,4)]. The same occured for actins [neuron (12,7)] and the α–chains of the class II HLA histocompatibility antigen subfamily [neuron (15,1)]. The other HLA histocompatibility antigen subfamilies were placed in the upper right corner of the map [class II β–chains in neurons (1,12) and (1,13) and class I in neurons (1,14) and (1,15)]. All hemoglobins were placed in two neighboring neurons [(1,7) and (1,8)]. The collagen family was splitted into two groups [ten sequences in neuron (7,15) and four in neurons (5,9) and (5,10)]. Myosin heavy chains [neurons (9,10), (9,11) and (10,11)], tropomyosins [neurons (8,9) and (9,9)] and keratins [neurons (7,10), (7,11) and (8,11)] were placed in neighboring neurons, close to other related proteins [lamins in neurons (8,11), (8,12) and (9,11); desmin and vimentin in neuron (8,11); desmoplakin in neuron (9,12); etc.]. Myosin light chains were placed somehow apart from this group [neurons (8,4), (8,5), (8,6) and (10,5)].

The resulting map should only be considered as one of many possible suitable ways to classify the set of proteins into a multiple number of clusters. Different learning protocols usually lead to similar, but not identical, maps.

Although the learning process is time consuming, it needs to be performed only once. Furthermore, once the network has self–organized itself, it can be used to rapidly classify new sequences. As an example of the retrieval stage, let us first consider the classification of one of the protein sequences that has been used to train the network: human interferon–α4b precursor (code name ina4$human in the SwissProt database). The unitary vector derived from the n first principal components of the bipeptidic matrix corresponding to this sequence is compared with the whole set of synaptic vectors to determine which is the neuron having the closest one. Thus, as a general result of the retrieval stage we obtain (Table 2): i) the position of the winner neuron, ii) the euclidian distance d between the input protein pattern and the synaptic vector of the winner neuron and iii) the list of learned proteins having that neuron as winner, with their corresponding distances. Comparing $d = 0.3480$ with the distances of that list we see that the input protein pattern has the same value as the human interferon α4b pattern. This coincidence suggests that the input protein is interferon–α4b (as in our case). However, it should be noted that, in general, as the vector space is not one–dimensional, two different vectors may have the same distance to a third one and the above analysis only gives a first guess for the searching process. The whole retrieval stage is very fast (about 15 SUN CPU–seconds).

Next, we fed the trained network with the horse interferon–α4 sequence. This sequence does not belong to the learning set that we used to train the network, but has a great homology with the corresponding human sequence [sequence identity = 75%, using the Needleman–Wunsch method (1970)]. As result of the retrieval stage the network also classified this sequence into the interferon–α family, though with a distance greater than those of human interferon–α sequences.

Discussion

For the particular set of 1758 human proteins analyzed in this work, the use of only 60 principal components seems enough to obtain a suitable classification. In fact, we have observed that the classification is somehow debased when too many principal components are considered. The reduction in the number of inputs of the network is close to the 69 variables taken into account by Van Heel for 10000 proteins.

Wu et al. (1992) have recently proposed another neural–network based method to classify protein sequences into families. The main difference between both ANN approaches resides on the type of learning procedure: Wu et al. have used a supervised learning algorithm, while we have used an unsupervised

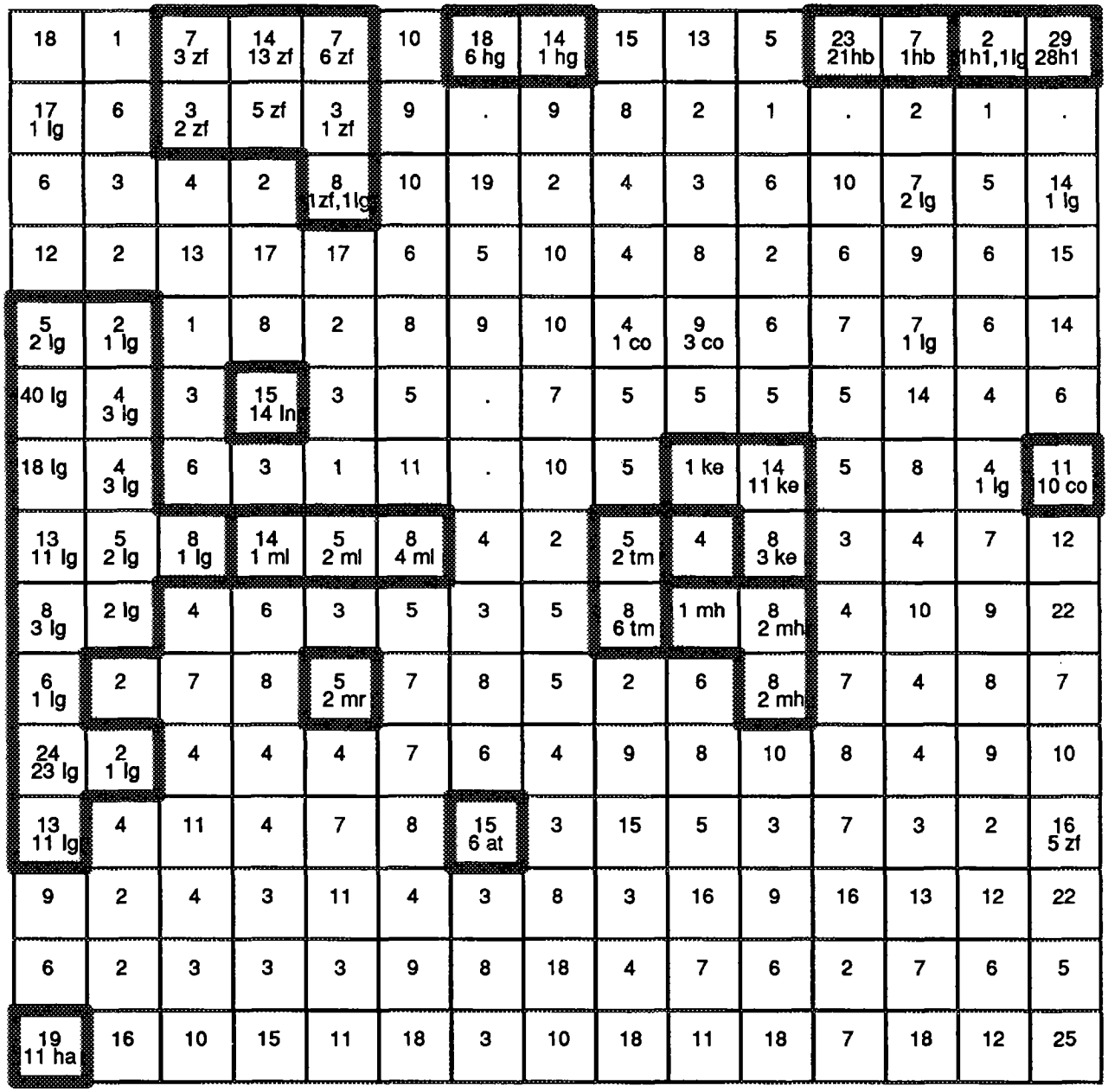

Fig.1: Topological map of 1758 human protein sequences. The network was trained using the first 60 principal components of the bipeptidic matrices as inputs. Learning proceeded during 30 epochs, linearly decreasing α each epoch ($\Delta t_\alpha = 1$), from a value of 0.9 ($a_1 = 0.08$ in the first 10 epochs and $a_2 = 0.0047619$ in the last 20) and decreasing the winner neighborhood every 2 epochs ($\Delta t_v = 2$). We only indicate the number of sequences having each neuron as winner and the positions of the following protein families: 6 actins (at), 14 collagens (co), 6 hemoglobins (hg), 29 HLA class I histocompatibility antigens (h1), 11 α–chains of HLA class II histocompatibility antigens (ha), 22 β–chains of HLA class II histocompatibility antigens (hb), 130 immunoglobulins (Ig), 14 interferon–α precursors (In), 15 keratins (ke), 7 myosin light chains (ml), 2 myosin regulatory light chains (mr), 5 myosin heavy chains (mh), 8 tropomyosins (tm) and 36 zinc–finger proteins (zf).

Winner neuron: (6,4)
Distance: 0.3480

Database name	Distance	Protein
inai$human	0.3208	Human Interferon-αi precursor
inaf$human	0.3315	Human Interferon-αf precursor
inaa$human	0.3318	Human Interferon-αa precursor
inac$human	0.3319	Human Interferon-αc precursor
ina7$human	0.3394	Human Interferon-α7 precursor
ina4$human	0.3480	Human Interferon-α4b precursor
inam$human	0.3495	Human Interferon-αm1 precursor
inad$human	0.3671	Human Interferon-αd precursor
inaw$human	0.4016	Human Interferon-αwa precursor
inab$human	0.4144	Human Interferon-αb precursor
ina8$human	0.4291	Human Interferon-α8 precursor
inak$human	0.4507	Human Interferon-αk precursor
inag$human	0.4647	Human Interferon-αg precursor
inah$human	0.5152	Human Interferon-αh precursor
ino1$human	0.5945	Human Interferon-α1 precursor

Table 2: Classification of human interferon-α4b using a network trained with 1758 human proteins.

one. They have trained several modules of a multi-layered network using the backpropagation algorithm. Each module was trained with known examples of a particular protein functional group (electron transfer proteins, oxidoreductases, transferases, hydrolases, etc.), using as inputs one or more "n-gram" encodings of the protein sequence. The known examples were taken from the annotated entries of the PIR protein sequence database, that is, those sequences that have been previously identified as belonging to a given protein superfamily. For example, they have trained one neural-network module with 383 entries corresponding to transferase sequences that belong to 157 superfamilies. When learning was accomplished, they tested the generalization ability of the trained network, using an independent set of annotated entries (116 transferase sequences). During learning, the synaptic efficacies between the neurons of different layers were changed in order to reduce a cost function. This function was a mesure of the difference between the actual outputs provided by the network to each entry and the corresponding desired outputs, that is, the outputs encoding the correct superfamily classification of the entries. Since the desired output values to each entry must be *a-priori* known, this kind of learning algorithm is called *supervised*. In fact, as the number and composition of protein families are not actually known, the application of a supervised method is not appropriate to classify the whole protein database and has to be restricted only to the annotated entries of the database. Instead, our approach can be extended, in principle, to the whole database, because we do not need to indicate *a-priori* which are the protein families.

The n-gram encoding of a given protein sequence, considered by Wu et al., is an n-dimensional matrix that gives the number of ocurrences of all possible patterns of n consecutives residues. They have analysed four different alphabets of residues: amino acids (20 symbols), exchange groups (6 symbols), structural groups (3 symbols) and hydrophobicity groups (2 symbols). Essentially, in both ANN approaches, the protein sequences are encoded in a similar way. In fact, our 20 x 20 protein representation is a particular case of n-gram, called bi-gram by Wu et al. (a2 in their notation). Interestingly, Wu et al. reported that the highest predictive accuracy and fastest convergence rate for their method are obtained when this particular encoding is concatenated with the amino acid compositions (i.e., the a1 n-gram) and some of the two or three lowest exchange group n-grams (the e1 and e2 or the e1, e2 and e3 n-grams). This suggests that our results may be further improved considering this kind of concatenated sequence representation.

In conclusion, the proposed unsupervised method can be a helpful computational tool for clustering proteins into families without having previous knowledge of the number and composition of the final clusters and for rapidly searching for sequence similarities in large

protein databases.

Acknowledgments

We thank D. Verrier for useful suggestions on our work and L.Hernandez for her valuable corrections of the manuscript.

References

Altschul, S.F.; Gish, W.; Miller, W.; Myers, E.W.; and Lipman, D.J. 1990. Basic local alignment search tool. *Journal of Molecular Biology* 215: 403–410.

Altschul, S.F.; and Lipman, D.J. 1990. Protein database searches for multiple alignments. *Proceedings of the National Academy of Sciences USA* 87: 5509–5513.

Andreassen, H.; Bohr, H.; Bohr, J.; Brunak, S.; Bugge, T.; Cotterill, R.M.J.; Jacobsen, C.; Kusk, P.; Lautrop, B.; Petersen, S.B.; Særmark, T.; and Ulrich, K. 1990. Analysis of the secondary structure of the human immunodeficiency virus (HIV) proteins p17, gp120, and gp41 by computer modeling based on neural network methods. *Journal of Acquired Immune Deficiency Syndromes* 3: 615–622.

Arrigo, P.; Giuliano, F.; Scalia, F.; Rapallo, A.; and Damiani, G. 1991. Identification of a new motif on nucleic acid sequence data using Kohonen's self–organizing map. *Computer Applications in the Biosciences* 7(3): 353–357.

Bengio, Y.; and Pouliot, Y. 1990. Efficient recognition of immunoglobulin domains from amino acid sequences using a neural network. *Computer Applications in the Biosciences* 6(4): 319–324.

Bohr, H.; Bohr, J.; Brunak, S.; Cotterill, R.M.J.; Lautrup, B.; Nørskov, L.; Olsen, O.H.; and Petersen, S.B. 1988. Protein secondary structure and homology by neural networks. The α–helices in rhodopsin. *FEBS Letters* 241(1,2): 223–228.

Bohr, H.; Bohr, J.; Brunak, S.; Cotterill, R.M.J.; Fredholm, H.; Lautrup, B.; and Petersen, S.B. 1990. A novel approach to prediction of the 3–dimensional structures of protein backbones by neural networks. *FEBS Letters* 261(1): 43–46.

Brunak, S.; Engelbrecht, J.; and Knudsen, S. 1990. Neural network detects errors in the assignment of mRNA splice sites. *Nucleic Acids Research* 18(16): 4797–4801.

Brunak, S.; Engelbrecht, J.; and Knudsen, S. 1991. Prediction of human mRNA donor and acceptor sites from the DNA sequence. *Journal of Molecular Biology* 220: 49–65.

Corpet, F. 1988. Multiple sequence alignment with hierarchical clustering. *Nucleic Acids Research* 16(22): 10881–10890.

Demeler, B.; and Zhou, G. 1991. Neural network optimization for *E.Coli* promoter prediction. *Nucleic Acids Research* 19(7): 1593–1599.

Engelbrecht, J.; Knudsen, S.; and Brunak, S. 1992. G+C–rich tract in 5' end of human introns. *Journal of Molecular Biology* 227: 108–113.

Farber, R.; Lapedes, A.; and Sirotkin, K. 1992. Determination of eukaryotic protein coding regions using neural networks and information theory. *Journal of Molecular Biology* 226: 471–479.

Ferrán, E.A.; and Ferrara, P. 1991a. Topological maps of protein sequences. *Biological Cybernetics* 65: 451–458.

Ferrán, E.A.; and Ferrara, P. 1991b. Unsupervised clustering of proteins. In *Artificial Neural Networks*, Proceedings of the First International Conference on Artificial Neural Networks, Vol.2, 1341–1344. Kohonen, T.; Mäkisara, K.; Simula, O.; and Kangas, J. eds. North–Holland.

Ferrán, E.A.; and Ferrara, P. 1992a. Clustering proteins into families using artificial neural networks. *Computer Applications in the Biosciences* 8(1): 39–44.

Ferrán, E.A.; and Ferrara, P. 1992b. A neural network dynamics that resembles protein evolution. *Physica A* 185: 395–401.

Ferrán, E.A.; and Ferrara, P. 1992c. A fast method to search for protein homologies using neural networks. In *Neural Networks: from biology to high energy physics*, Proceedings of the Second Elba Workshop. Forthcoming.

Ferrán, E.A.; Pflugfelder, B.; and Ferrara, P. 1992. Large–scale application of neural networks to protein classification. In *Artificial Neural Networks 2*, Proceedings of the Second International Conference on Artificial Neural Networks, Vol.2, 1521–1524. Aleksander, I.; and Taylor, J. eds. North–Holland.

Ferrán, E.A.; and Ferrara, P. 1993. Self-organized neural maps of human protein sequences. Submitted for publication.

Ferrán, E.A.; and Pflugfelder, B. 1993. A hybrid method to cluster protein sequences based on statistics and artificial neural networks. Submitted for publication.

Frishman, D.; and Argos, P. 1992. Recognition of distantly related protein sequences using conserved motifs and neural networks. *Journal of Molecular Biology* 228: 951–962.

Gribskov, M., McLachlan, A.D.; and Eisenberg, D. 1987. Profile analysis: detection of distantly related proteins. *Proceedings of the National Academy of Sciences USA* 84: 4355–4358.

Hayward, S.; and Collins, J.F. 1992. Limits on α–helix prediction with neural network models. *Proteins: Structure, Function, and Genetics* 14: 372–381.

Hirst, J.D.; and Sternberg, M.J.E. 1991. Prediction of ATP–binding motifs: a comparison of a

perceptron–type neural network and a consensus sequence method. *Protein Engineering* 4(6): 615–623.

Hirst, J.D.; and Sternberg, M.J.E. 1992. Prediction of structural and functional features of protein and nucleic acid sequences by artificial neural networks. *Biochemistry* 31(32): 7211–7218.

Holbrook, S.R.; Muskal, S.M.; and Kim, S–H. 1990. Predicting surface exposure of amino acids from protein sequence. *Protein Enginnering* 3(8): 659–665.

Holley, L.H.; and Karplus, M. 1989. Protein secondary structure prediction with a neural network. *Proceedings of the National Academy of Sciences USA* 86: 152–156.

Horton, P.B.; & Kanehisa, M. 1992. An assessment of neural network and statistical approaches for prediction of *E. Coli* promoter sites. *Nucleic Acids Research* 20(16): 4331–4338.

Hotelling, H. 1933. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* 24: 417–441, 498–520.

Kneller, D.G.; Cohen, F.E.; and Langridge, R. 1990. Improvements in protein secondary structure prediction by an enhanced neural network. *Journal of Molecular Biology* 214: 171–182.

Kohonen, T. 1982. Self–organized formation of topologically correct feature maps. *Biological Cybernetics* 43: 59–69.

Ladunga, I.; Czakó, F.; Csabai, I.; and Geszti, T. 1991. Improving signal peptide prediction accuracy by simulated neural network. *Computer Applications in the Biosciences* 7(4): 485–487.

Lapedes, A.; Barnes, C.; Burks, C.; Farber, R.; and Sirotkin, K. 1990. Application of neural networks and other machine learning algorithms to DNA sequence analysis. In *Computers and DNA*, 157–182. Bell, G.; and Marr, T. eds. SFI Studies in the Sciences of Complexity, vol VII: Addison–Wesley.

Le Cun, Y. 1985. A learning scheme for asymmetric threshold networks. In Proceedings of Cognitiva, 599–604.

Lipman, D.J.; and Pearson, W.R. 1985. Rapid and sensitive protein similarity searches. *Science* 227: 1435–1441.

Lukashin, A.V.; Anshelevich, V.V.; Amirikyan, B.R.; Gragerov, A.I.; & Frank-Kamenetskii, M.D. 1989. *Journal of Biomolecular Structure & Dynamics* 6(6): 1123–1133.

Maddox, J. 1992. Ever–longer sequences in prospect. *Nature* 357: 13.

McGregor, M.J.; Flores,T.P.; and Sternberg, M.J.E. 1989. Prediction of β–turns in proteins using neural networks. *Protein Engineering* 2(7): 521–526.

Muskal, S.M.; Holbrook, S.R.; and Kim, S–H. 1990. Prediction of the disulfide–bonding state of cysteine in proteins. *Protein Enginnering* 3(8): 667–672.

Muskal, S.M.; and Kim, S–H. 1992. Predicting protein secondary structure content. A tandem neural network approach. *Journal of Molecular Biology* 225: 713–727.

Nakayama, S–I.; Shigezumi, S.; and Yoshida, M. 1988. Method for clustering proteins by use of all possible pairs of amino acids as structural descriptors. *Journal of Chemical Information and Computer Sciences* 28: 72–78.

Needleman, S.B.; and Wunsch, C.D. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology* 48: 443–453.

O'Neill, M.C. 1991. Training back–propagation neural networks to define and detect DNA–binding sites. *Nucleic Acids Researchs* 19(2): 313–318.

O'Neill, M.C. 1992. *Escherichia Coli* promoters: neural networks develop distinct descriptions in learning to search for promoters of different spacing classes. *Nucleic Acids Researchs* 20(13): 3471–3477.

Pearson, K. 1901. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine* 6(2): 559–572.

Petersen, S.B.; Bohr, H.; Bohr, J.; Brunak, S.; Cotterill, R.M.J.; Fredholm, H.; and Lautrup, B. 1990. Training neural networks to analyse biological sequences. *Trends in Biotechnology* 8: 304–308.

Pflugfelder, B.; and Ferrán, E.A. 1992. Bipeptidic matrix clustering. In Proceedings of the SAS European Users Group International Conference, 656–686. SAS Institute.

Qian, N.; and Sejnowski, T.J. 1988. Predicting the secondary structure of globular proteins using neural network models. *Journal of Molecular Biology* 202: 865–884.

Rao, R. 1960. Multivariate analysis: an indispensable aid in applied research. *The Indian Journal of Statistics (Sankhyã)* 22: 317–338.

Rao, R. 1964. The use and interpretation of Principal Component Analysis in applied research. *Sankhyã* Series A, 26: 329–358.

Rose, V.S.; Croall, I.F.; and MacFie, H.J.H. 1991. An application of unsupervised neural network methodology (Kohonen topology–preserving mapping) to QSAR analysis. *Quantitative Structure–Activity Relationships* 10: 6–15.

Rosenblatt, F. 1962. *Principles of Neurodynamics.* Spartan Books, New York.

Rost, B.; and Sander, C. 1992. Jury returns on structure prediction. *Nature* 360: 540.

Rumelhart, D.E.; Hinton, G.E.; and Williams, R.J. 1986. Learning representations by back–propagating errors. *Nature* 323: 533–536.

SAS Institute Inc. 1985. *SAS User's Guide: statistics, version 5 Edition.* Cary, NC: SAS Institute, Inc.

Sasagawa, F.; and Tajima, K. 1993. Prediction of protein secondary structures by a neural network. *Computer Applications in the Biosciences* 9: 147–152.

Snyder, E.E.; & Stormo, G.D. 1993. Identification of coding regions in genomic DNA sequences: an application of dynamic programming and neural networks. *Nucleic Acids Researchs* 21(3): 607–613.

Stolorz, P.; Lapedes, A.; and Xia, Y. 1992. Predicting protein secondary structure using neural net and statistical methods. *Journal of Molecular Biology* 225: 363–377.

Stormo, G.D.; Schneider, T.D.; Gold, L.; and Ehrenfeucht, A. 1982. Use of the 'Perceptron' algorithm to distinguish translational initiation sites in *E.Coli*. *Nucleic Acids Research* 10(9): 2997–3011.

Sulston, J.; Du, Z.; Thomas, K.; Wilson, R.; Hillier, L.; Staden, R.; Halloran, N.; Green, P.; Thierry-Mieg, J.; Qiu, L.; Dear, S.; Coulson, A.; Craxton, M.; Durbin, R.; Berks, M.; Meltztein, M.; Hawkins, T.; Ainscough, R.; and Waterston, R. 1992. The *C. elegans* genome sequencing project: a beginning. *Nature* 356: 37–41.

Uberbacher, E.C.; and Mural, R.J. 1991. Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proceedings of the National Academy of Sciences USA* 88: 11261–11265.

Van Heel, M. 1991. A new family of powerful multivariate statistical sequence analysis techniques. *Journal of Molecular Biology* 220: 877–887.

Vieth, M.; and Koliński, A. 1991. Prediction of protein secondary structure by an enhanced neural network. *Acta Biochimica Polonica* 38(3): 335–351.

Vieth, M.; Koliński, A.; Skolnick, J.; and Sikorski, A. 1992. Prediction of protein secondary structure by neural networks - Encoding short and long range patterns of amino acid packing. *Acta Biochimica Polonica* 39(4): 369–392.

von Heijne, G. 1991. Computer analysis of DNA and protein sequences. *European Journal of Biochemistry* 199: 253–256.

Wade, R.C.; Bohr, H.; and Wolynes, P.G. 1992. Prediction of water binding sites on proteins by neural networks. *Journal of the American Chemical Society* 114: 8284–8285.

Watson, J.D. 1990. The human genome project: past, present and future. *Science* 248: 44–49.

Wu, C.; Whitson, G.; McLarty, J.; Ermongkonchai, A.; and Chang, T. 1992. Protein classification artificial neural system. *Protein Science* 1: 667–677.

Zhang, X.; Mesirov, J.P.; and Waltz, D.L. 1992. Hybrid system for protein secondary structure prediction. *Journal of Molecular Biology* 225: 1049–1063.