2001 Proceedings of the 23rd Annual EMBS International Conference, October 25-28, Istanbul, Turkey

THREE DIMENSIONAL REPRESENTATION OF AMINO ACID CHARACTERISTICS

O.U. Sezerman[1], R. Islamaj[2], E. Alpaydin[2]
[1]Laborotory of Computational Biology, Sabancı University, Istanbul, Turkey.
[2] Computer Engineering Department, Bogazici University, Istanbul, Turkey.

Abstract-**Amino acid substitution matrices which shows the similarity scores between pairs of amino acids have been widely used in protein sequence alignments. These matrices are based on the Dayhoff model of evolutionary substitution rates. Using machine learning techniques we obtained three dimensional representations of these matrices while preserving most of the information obtained in the matrices. Vector representation of amino acids has many applications in pattern recognition.**
***Keywords - substitution* matrices, machine learning, distance mapping.**

I. INTRODUCTION

Protein similarity score matrices are constructed from substitution matrices which are obtained from multiple alignment of several evolutionally related sequences[1-5]. The substitution matrices are derived from evolutionary amino acid substitution frequencies of protein sequences. Using information theory, these frequencies are converted into similarity scores. These scores are correlated with the physical and chemical properties of amino acids. These matrices are used in protein sequence comparison, generating sequence profiles and in database searches for similar sequences. They are also used in sequence and structural pattern recognition problems such as secondary structure prediction and finding contact maps of proteins.

The aim of this work is to simplify the representation of the twenty amino acids in a metric space with minimum loss of information. What is required is a mapping, y = f (x), where the input value will be the similarity score from a given matrix and the output value should be a multi-dimensional vector for each amino acid.

Since only the x values are known, the problem can be viewed as an unsupervised learning problem. Machine learning techniques offer several alternatives to resolve the problem. The typical approach is to refine iteratively the representation of symbols into multi-dimensional space by minimising the error for the obtained vectors that correspond to the given similarity scores.

For this problem, usage of artificial neural networks [6,7] may be practical, since neural network learning methods provide a robust approach to approximating real and vector-valued functions, which is exactly the case in this problem. The fact that the errors in training examples are tolerable is another advantage that makes neural networks convenient, because the score matrices are not guaranteed to be reducible to a space of given dimensions.

On the other hand, usage of hybrid techniques that combine the virtues of mathematical verities and machine learning techniques is also possible. The score matrix can be put into a form that more readily reflects an N-dim space nature, then this new form of representation can be used to obtain the reduced number of dimensions through a refinement cycle that converges to the target function.

Instead of going from similarity matrix into a metric space and then into vector space, we can directly go from similarity score matrices to multidimensional space using nonlinear mapping techniques. This way we eliminate the risk of loosing information during the similarity to distance transformation stage.

II. METHODOLOGY

Linear mapping stage takes the similarity values as input and uses a transformation to map the amino acids on the metric space relying on the intuitive result that distance and similarity are opposite concepts. So the inverse of similarity should define distance, meaning that the higher the similarity score, the closest the symbols are expected to be positioned in space.

To transform similarity scores to distance values, several formulas are used, out of which the ones that displayed a better performance were chosen. The performance decision was based on the weight ratio of the greatest N eigenvalues to the 20 eigenvalues obtained from the distance matrix for N dimensions. The greater the value of the ratio, the higher the amount of the original information that is conserved during this transformation. The top three performing distance formulas are

$$D_1(i,j) = 1 / (S_{i,j} - \min S + \text{offset})^2$$
$$D_2(i,j) = 1 / \sqrt{(S_{i,j} - \min S + \text{offset})^3}$$
$$D_3(i,j) = 1 / \sqrt{((S_{i,j} - \min S)^2 + \text{offset})}$$

Where $S_{i,j}$ is the similarity score between amino acids i and j, and minS is the minimum score in the matrix. Since the scores can be negative and distances have to be positive, we subtracted the minimum score in the matrix from all the scores in the matrix. Then the offset value is added to overcome the division by zero error.

To have a fair representation in a metric space, all distances should conform the triangular inequality rule.

For any i, j and k

$$D_{ij} \le D_{ik} + D_{jk}$$

A linear mapping requires that this condition hold for any triple distance values. Since the offset value is the only variable in the equations, we tried different values for the given scoring matrix until the triangular inequality rule is conformed.

After the linear mapping of amino acids into a metric space, we calculate the eigenvalues and the eigenvectors of the distance matrix. 6. To map the data symbols into N-dimensional space, eigenvectors of the greatest N eigenvalues are used. Therefore the greatest eigenvector multiplied by its corresponding eigen value is the x-coordinate, the sec-

ond greatest eigenvector multiplied by its eigenvalue is the y-coordinate of the symbols and so on. The eigenvectors are the new coordinate axes. Thus, we obtain the initial distance vectors for the amino acids on the N dimensional space defined by those eigenvectors.

These initial distance vectors are then subjected to an iterative refinement procedure. In this procedure, we first calculate the distance (L_{ij}) between amino acid i and j in the N-dimensional space. Next, we calculate the total error of transforming into N dimensions as $\sum\sum (L_{ij}-D_{ij})^2$, where Dij is the actual distance between amino acid i and j. The vectors are updated so that the error is minimised. This procedure continues until the vector coordinates converge for all the amino acids.

While transforming the similarity matrix into distance matrix, we loose some information. Similarity of each amino acid to itself is different for each amino acid, depending on their observed substitution frequencies orchestrated via evolution. On the other hand, ideally the distance of an amino acid to itself should be zero, but this is not possible when we have different self-similarity scores.

We avoided this problem by using encoding-decoding technique. The main motive was the need for a nonlinear method so that distances were not needed. We can use directly the similarity scores. Second, a transformation was desired with the property that it would compress the data and the data could be obtained back with minimum loss and at a minimum cost. The encoding-decoding method described below satisfies both conditions: It is nonlinear and its cost is the time required to train the multilayer perceptron which contains 2, 3 or 4 hidden units depending on the number of dimensions we choose to explain the symbols.

As seen in Fig.1, the algorithm takes the similarity scores and the dimension of the vector space as the input values. After initializing the weights and the learning rate, the algorithm calculates the coordinates as the sum of the products of encoding weights and the similarity scores. Output values are determined as the sum of the products of decoding weights and the coordinates. The error is the difference between the input and the output values of the amino acids. We then take partial derivatives of the error function with respect to the encoding and decoding weights in order to minimise the error function. Then update the weights and the learning rate factor. We continue this procedure until the vector representation for each amino acid converges.

III. RESULTS

There are several similarity score matrices that represent different evolutionary relations and are obtained by using different statistical measures. In this work, we used four matrices; BLOSUM 45[1], BLOSUM 60, PAM 250 and Dayhoff[4] matrix. We used several distance measures in mapping the similarity scores into the metric space. The mapping result obtained from top three-distance measure is summarised in Table 1-3. On the following tables (a) stands for offset values, (b) is number of triples that the triangular inequality does not hold, (c) is the ratio of the sum of the greatest 3 eigenvalues to the sum of all eigenvalues.

The most successful distance measure for three dimensional representation of amino acids was D_3 since it has preserved the highest percentage of the similarity score information for all the similarity score matrices (Approximately 60%) except BLOSUM45 Table 3.

TABLE 1. Mapping with the formula D1 = 1 / (Si,j - minS + offset)^2

Blosum45			Pam250			Blosum60			Dayhoff		
a	b	c	a	b	c	a	b	c	a	b	c
8	20	52	16	8	57	7	14	58	10	76	60
9	8	52	19	2	55	8	6	57	11	52	60
10	6	51	20	0	54	9	0	57	16	8	58
12	0	51	22	0	52	10	0	56	20	0	55

TABLE 2. Mapping with the formula D2 = 1 / √ (Si,j - minS + offset)^3

Blosum45			Pam250			Blosum60			Dayhoff		
a	b	c	a	b	c	a	b	c	a	b	c
6	8	52	9	22	57	7	14	58	10	10	58
7	4	52	11	8	57	8	6	57	12	2	56
8	0	51	13	2	57	9	0	57	14	8	54
9	0	51	14	0	56	10	0	56	15	0	53

TABLE 3. Mapping with the formula D3 = 1 / √ ((Si,j – minS)^2+offset)

Blosum45			Pam250			Blosum60			Dayhoff		
a	b	c	a	b	c	a	b	c	a	b	c
10	2	51	20	8	59	6	2	59	17	10	60
14	0	52	22	0	59	7	0	58	22	0	59

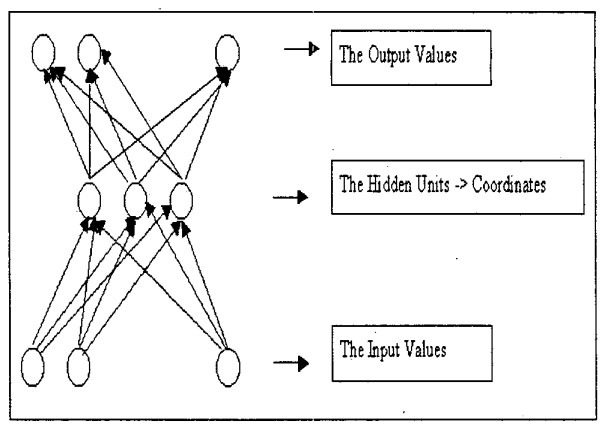

Figure 1 Architecture of encoding decoding technique

If we map the amino acids into four dimensional space percentage of the preserved information increases by 7% across the board. The results obtained from the four dimensional representation of D_3 are summarised in Table 4. The best results are obtained from Pam250 and Dayhoff matrices even though the difference is marginal.

The results obtained from Pam 250 are shown in Fig.2. We see clustering of hydrophobic residues within this cluster we see additional clustering of aromatic residues. Charged and polar amino acids also form a cluster. The third cluster is formed by small aliphatic amino acids. In evolutionary data we observe accepted mutations within these clusters. As can be seen in Fig.2 the amino Cystine stands alone. This is expected since cystine is the only amino acid that can form disulfide bond, which is one of the most important stabilising factors for the protein structure. Therefore Cystine does not like to be substituted by other amino acids since they are the functional sites of proteins.

As expected increasing the dimensionality decreases the amount of lost information. The sum of the information content of other dimensions is only 23% of the total. Going into the 5th dimension improves the result by a few percentage points in each case. There is always loss of some information in going from similarity to distance and reducing the dimensionality introduces new additional loss.

To overcome this problem we use the encoding decoding technique. This technique performs direct mapping from similarity score into N-dimensional space. Since this method converges to a local minima, we perform several runs. Blosum 60 similarity score matrix gave the best clustering of the amino acids in all the runs. Even in two dimensions Blosum 60 forms the expected clusters that are intrinsic in evolution Fig.3. But we cannot observe the distinction of Cystine from the other amino acids in two dimensions. Adding the third dimension separates the cystine from the other cluster Fig.4.

TABLE 4. 4-D imensional values

	Offset Value	Triangular Inequality	Eigenvalues Info
Blosum45	14	0	59.2
Pam250	22	0	66.3
Blosum60	7	0	64.7
Dayhoff	22	0	66.9

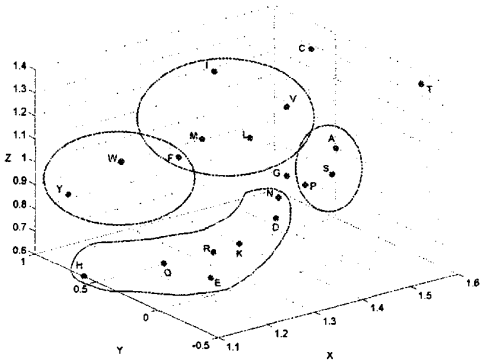

Figure 2. Blosum 60 with distance refinement

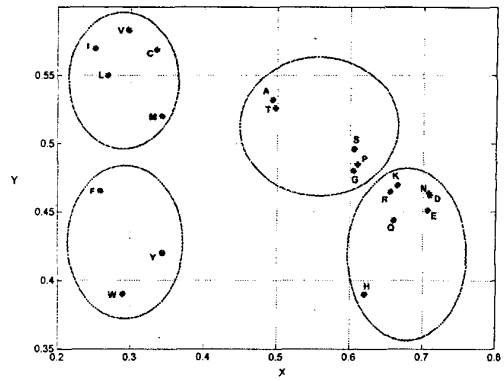

Figure 3. Blosum 60 using encoding decoding

IV. Discussion

In this work we studied three methods to decrease the dimensionality of the similarity score matrices. Linear mapping always gave consistent results bot information was lost while converting the similarity to distance. Iterative distance refinement approach uses a different initial condition that's why better mapping could be as a result of better search of the solution space. But there is the problem of convergence in this approach, in some of the matrices amino acid position vectors did not converge satisfactorily.

Encoding decoding approach gave the best results, because of the direct transformation from similarity to metric space. This method also starts from several initial states so it searches the space more efficiently, but as in the distance refinement approach it falls into local minima and it has problems of convergence. But it converged faster than the distance refinement approach.

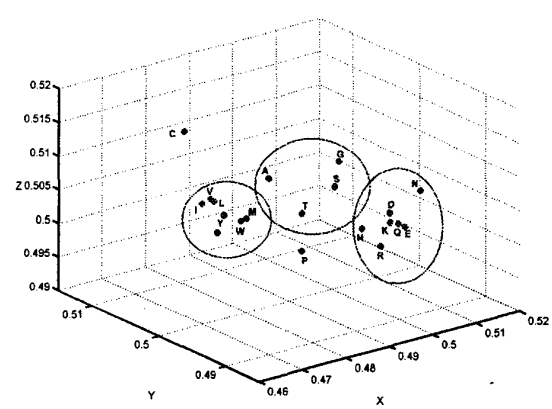

Figure 4. Blosum60 using encoding decoding in 3-D

V. Conclusion

In this work we successfully applied machine learning techniques to the problem of reducing the dimensionality of the amino acid similarity score matrices. Representing amino acids as three dimensional vectors enables us to use these vectors as the input instead of the letter code of amino acids, while trying to find some sequence or structural patterns of protein sequences. The idea is that these vectors preserve all the necessary evolution information intrinsic in the similarity score matrices. When we use these vectors instead of the symbols we do not need to provide additional information such as evolutionary data, size, hydrophobicity etc. Since all these information are imbedded in the vector representations. This simplifies the problem immensely.

When we transform the similarity data into distance data some information is lost. Therefore, neural network implementation yielded better results. The distances between the clusters and within the cluster were minimum in this approach. Although deviations to a tolerable degree were observed, the clusters represented the actual groupings of the amino acids represented in the score matrices.

References

[1] S. Henikoff and J.G. Henikoff, "Amino acid substitution matrices from protein blocks," *PNAS,* vol.89, pp10915-109191, November 1992.

[2] A. D. Maclachlan, " Test for comparing related amino acid sequences," *J. Mol. Biol.*, 1971, vol.61, pp.409-424.

[3] D. Feng, M.S. Johnson, and R. F. Doolittle, "Aligning amino acid sequences: comparison of commonly used methods," *J. Mol. Evo.*, 1985, vol.21, pp.121-125.

[4] M. O. Dayhoff, R. M. Schwartz and B. C. Orcutt, " A model of evolutionary change in proteins," *Atlas of protein sequence and structure,* 1973, vol.5, pp.345-352.

[5] S. F. Altschull, "Amino acid substitition matrices from an information theoretic perspective," *J. Mol. Biol.,*1991, vol. 219, pp. 555-565.

[6] J. A. Anderson, " An introduction to neural networks," Cambridge Mass, MIT Press, 1965.

[7] S. Grossberg, "Neural Networks: From foundations to applications," Short-Course notes, Boston University, Boston, Mass., May 6-11, 1990.