Clustering and Visualization of Large Protein Sequence Databases by Means of an Extension of the Self-Organizing Map

Panu Somervuo and Teuvo Kohonen

Helsinki University of Technology
Neural Networks Research Centre
P.O.Box 5400, FIN-02015 HUT, Finland
panu.somervuo@hut.fi, teuvo.kohonen@hut.fi

Abstract. New, more effective software tools are needed for the analysis and organization of the continually growing biological databases. An extension of the Self-Organizing Map (SOM) is used in this work for the clustering of all the 77,977 protein sequences of the SWISS-PROT database, release 37. In this method, unlike in some previous ones, the data sequences are not converted into histogram vectors in order to perform the clustering. Instead, a collection of true representative model sequences that approximate the contents of the database in a compact way is found automatically, based on the concept of the generalized median of symbol strings, after the user has defined any proper similarity measure for the sequences such as Smith-Waterman, BLAST, or FASTA. The FASTA method is used in this work. The benefits of the SOM and also those of its extension are fast computation, approximate representation of the large database by means of a much smaller, fixed number of model sequences, and an easy interpretation of the clustering by means of visualization. The complete sequence database is mapped onto a two-dimensional graphic SOM display, and clusters of similar sequences are then found and made visible by indicating the degree of similarity of the adjacent model sequences by shades of gray.

1 Introduction

The amount of DNA sequences, protein sequences, and molecule structures studied and reported, e.g., in the Internet is already overwhelming. One should develop better tools for the analysis of the existing databases. Thereby, however, it will also become possible to make new discoveries, without the need to carry out the real biological and chemical experiments.

Among the new challenges one may mention finding the hidden relations between the data items, revealing structures from large databases, and representing the results to the human in a comprehensible way. The classification and clustering of the sequences may reveal new unknown connections between them. The visualization of large data sets in a compact way may give insights into the data and lead to the development of new ideas and theories.

Although the data mining applications in general require very specialized and tailored solutions, it is interesting to note that some general principles and methods can already define a framework for these tasks. One such method is the Self-Organizing Map (SOM) [11, 13, 14, 16]. It is a clustering and visualization tool, which has been applied to a diversity of problems. This paper points out the potential of the SOM in the clustering and organization of large sequence databases.

The SOM has already been applied to the clustering of protein sequences. In [6], the sequences were converted into 400-dimensional dipeptide histogram vectors. In [7], similar amino acids were grouped together before computing the histogram vectors. In [9], the sequences were converted into vectors by fractal encoding. Before that the sequences were aligned. In [2], each position of the sequence was represented as a 20-dimensional vector; each vector component corresponded to one amino acid. The whole sequence was then converted into an L-by-20-dimensional vector, where L is the length of the global alignment of all sequences. As a conclusion, in all these works the data has been encoded by vectors before feeding to the SOM.

A new method suggested by Kohonen [15], however, allows the organization of nonvectorial data items, too. The clustering and organization of the sequence database can therefore be based on any user-defined algorithm, e.g. Smith-Waterman [20], BLAST [1], or FASTA [19]. In the present work, the FASTA method was used for computing the sequence similarities. The SOM was then applied to clustering all the 77,977 protein sequences of the SWISS-PROT database, release 37 [3].

2 The Self-Organizing Map for both vectorial and nonvectorial data

In its original form the Self-Organizing Map is a nonlinear projection method that maps a high-dimensional metric vector space, or actually only the manifold in which the vectorial samples are located, onto a two-dimensional regular grid in an orderly fashion [11, 14]. The SOM differs from the traditional projection methods such as multidimensional scaling, MDS [17] in that unlike in the latter, each original sample is not represented separately, but a much smaller set of *model vectors*, each of the latter associated with one of the grid nodes, is made to approximate the set of original samples. The SOM thus carries out a kind of vector quantization, VQ [8], in which, however, the model vectors (called codebook vectors in VQ) may be imagined to constitute the nodes of a flexible, smooth network that is fitted to the manifold of the samples.

The SOM principle is not restricted to metric vector spaces, however. It has been pointed out by one of the authors [15] that any set of items, for which a similarity or distance measure between its elements can be defined, can be mapped on the SOM grid in an orderly fashion. This is made possible by the following principle, which combines the concept of the generalized median of a set [12] with the batch computation principle of the SOM [14].

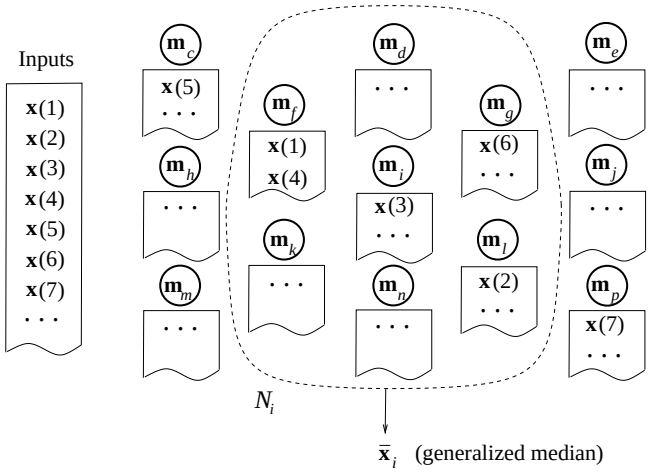

Fig. 1. Illustration of the SOM algorithm for nonvectorial data. Each of the input items $\mathbf{x}(1), \mathbf{x}(2), \ldots$ is copied into the sublist under that model that has the smallest distance from the respective input item. After that, the generalized median $\bar{\mathbf{x}}_i$ in each neighborhood set N_i is determined, and the old value, say $\mathbf{m}_i$, is replaced by $\bar{\mathbf{x}}_i$. This cycle is repeated from the beginning as many times as the models are not changed any longer.

Let us concentrate on the special SOM that is able to map nonvectorial items. Consider Fig. 1 in which a regular grid is shown, with some general model $\mathbf{m}_c \ldots \mathbf{m}_p$ associated with each grid node. Assume that a sublist that contains a subset of input items $\mathbf{x}(i)$ can be associated with each model. Each of the input items $\mathbf{x}(1), \mathbf{x}(2), \ldots$ is compared with all the models and listed under that one that has the smallest distance from the respective input item. The $\mathbf{x}(1), \mathbf{x}(2), \ldots$ will thus be distributed under the closest models.

Define for each model, say $\mathbf{m}_i$, a neighborhood set N_i (the set of models located within a certain radius from the node i in the grid). Consider the union of all the sublists within N_i (shown by the set line in Fig. 1) and try to find the "middlemost" input sample in N_i. This sample is called the *generalized median* of N_i, and it is defined to be identical with the input sample that has the smallest sum of distances from all the other samples of N_i.

In forming the sum of distances, the contents of the sublists within N_i can be *weighted* so that the weight is a function of the distance of the nodes of the grid from, say, node i. This corresponds to the neighborhood function used with the traditional SOM [14].

Comment 1. If the input samples had been real scalars and the distance measure were the absolute value of their difference, it is easy to show that the "generalized median" coincides with the arithmetic median.

Comment 2. If the input samples were real vectors, and the distance measure were Euclidean, and if the item with the smallest sum of the *squares* of

distances from the other items were sought, the "generalized median" would coincide with the arithmetic mean of the union of the lists. In this case the "median" is not restricted to the input samples, but belongs to the same domain.

For each N_i in Fig. 1, $i = c, d, \ldots, p$ the generalized median is now determined, and the old models $\mathbf{m}_c \ldots \mathbf{m}_p$ are replaced by the respective generalized medians.

After this replacement, the original models have now been changed, and if the same input samples are compared with them, they are now redistributed in a different way in the lists. Eventually, however, in a finite number of iterations of this type the process will converge, after which the models approximate the input samples in an orderly fashion.

It is not yet mathematically proven that the above process converges, at least into a unique equilibrium. In practice, convergence means that the lists will not be changed any longer in further iterations. Furthermore, there may exist alternative states into which the map may converge. A proof of a similar "batch map" process with vectorial items has been presented [4], but any conclusions for nonvectorial items can only be drawn from the experimental results, for which no problems have so far existed.

Comment 3. Like in the traditional SOM for vectorial items [14], the radius of the neighborhood set N_i in the beginning of the process may be selected as fairly large and put to shrink monotonically in further iterations. The speed of shrinking should be determined experimentally so that the global ordering is achieved.

3 Clustering of 77,977 protein sequences

The SWISS-PROT database, release 37 (12/98) [3] consists of 77,977 protein sequences. The sequences contain altogether 28,268,293 amino acid residues. Organization of a database of this size, and representation of the result in a compact form is a challenging task. Our purpose was to use the definition of distances between the protein sequences, as made in the FASTA method [19] for the computation of the SOM as described in the previous section. A 30-by-20 SOM size was chosen.

The convergence of the nonvectorial SOM algorithm is safer and faster, if the initial models are already two-dimensionally ordered, roughly at least, although not yet optimized. In a couple of earlier works [6, 7], protein sequences were ordered according to the similarity of their dipeptide histograms. We found this method useful for the definition of a rough initial order to the SOM. Then, however, extra auxiliary *model vectors* have to be introduced and associated with the nodes. The initial ordering of the vectorial models in this auxiliary SOM proceeded in the traditional way. Each map node was provided with a 400-dimensional model vector, each component of which was initialized with a random value between zero and unity, whereafter the vectors were normalized to unit length. Training was made by the 400-dimensional dipeptide histograms

using 30 batch cycles. A Gaussian neighborhood kernel, the standard deviation of which decreased linearly from 30 to 1 during training, was used.

Next the nodes were labeled by those protein sequences that represented the medians in the sublists under the respective nodes (cf. Fig. 1). When this labeling was ready, the vectorial parts of the models could be abandoned, and the ordering could be continued by the method described in Sec. 2.

After this initializing phase, the true protein sequences were used as inputs as described in Sec. 2 and the winner nodes were determined by the FASTA method. The source code for the FASTA computation was extracted from the FASTA program package, version 3.0 [18]. The parameter *ktup* was set to 2, the amino acid substitution scores were taken from the BLOSUM50 matrix, and the final optimized score for the sequence similarity was computed by dynamic programming.

The SOM was trained for twenty batch cycles, using the neighborhood radius of one. (Since the SOM was already ordered, there was no need to use a shrinking kernel any longer.) Since the sequence similarities instead of their distances were finally computed, for the "median" we had to take that sequence in the union of the neighboring sublists that had the largest sum of similarity values with respect to all the other sequences in the neighboring lists. The Gaussian neighborhood function was applied for the weighting of the similarities.

It would have presented a very high computing load to the algorithm if all the 77,977 protein sequences had been used as inputs at each batch computation cycle. The computing load could be reduced to less than ten percent, without essentially deteriorating the (statistical) accuracy of the batch computation, by randomly picking up 6,000 sample sequences from the 77,977 ones for each batch cycle. After 20 such sampled training cycles, one final training cycle was carried out using all the available sequences as the inputs.

The resulting SOM is shown in Fig. 2. The map nodes have been labeled according to the identifiers of the final prototypes that resulted in the "median map" method.

For comparison, another labeling was carried out by listing all data sequences under the best-matching nodes and then performing the majority voting for each list according to the PROSITE classes, release 15 [10] of the sequences. This result is shown in Fig. 3. Since the PROSITE database did not give any class for 37,743 sequences of the SWISS-PROT database, the PROSITE label of the node does not necessarily characterize all sequences of the node.

The clusters can be characterized by means of the known protein families. Those classes whose members are strongly similar are mapped to small areas on the map, while other classes may be spread more widely. Actins and rubisco-large are examples of the classes which form sharp areas on the map. Globin is a large family which is composed of subfamilies. The globin sequences are mostly mapped on the top-left corner of the SOM. Hemoglobin beta chains are represented on the corner, hemoglobin alpha chains are in the cluster below catalases, and myoglobins are located below hemoglobin alpha chains. One sharp cluster on the top of the map consists of efactor-gtp sequences. Between globins

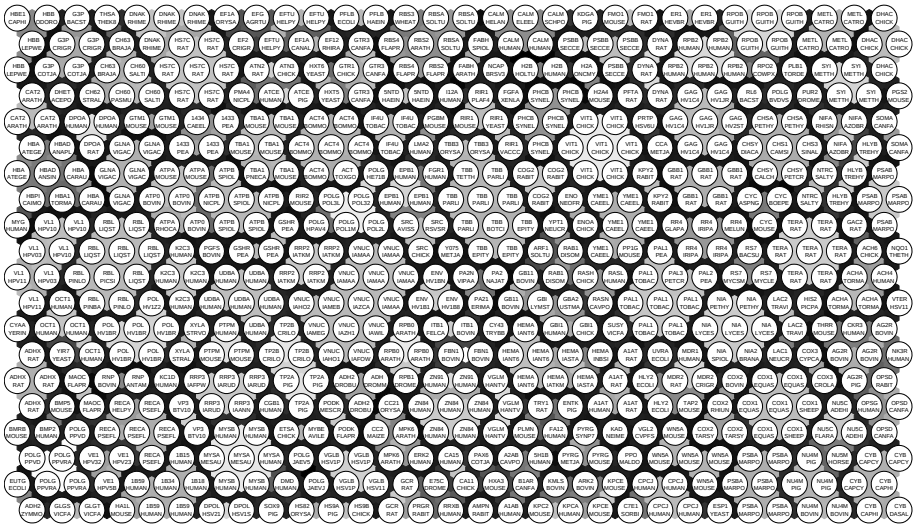

Fig. 2. A 30-by-20-unit hexagonal SOM grid. The SOM was constructed using all the 77,977 protein sequences of the SWISS-PROT release 37. Each node contains a prototype sequence and a list of data sequences. The labels on the map nodes are the SWISS-PROT identifiers [3] of the prototype sequences. The upper label in each map node is the mnemonic of the protein name and the lower label is the mnemonic of the species name. The similarities of the neighboring prototype sequences on the map are indicated by shades of gray. The light shades indicate a high degree of similarity, and the dark shades a low degree of similarity, respectively. Light areas on the map reveal large clusters of similar sequences.

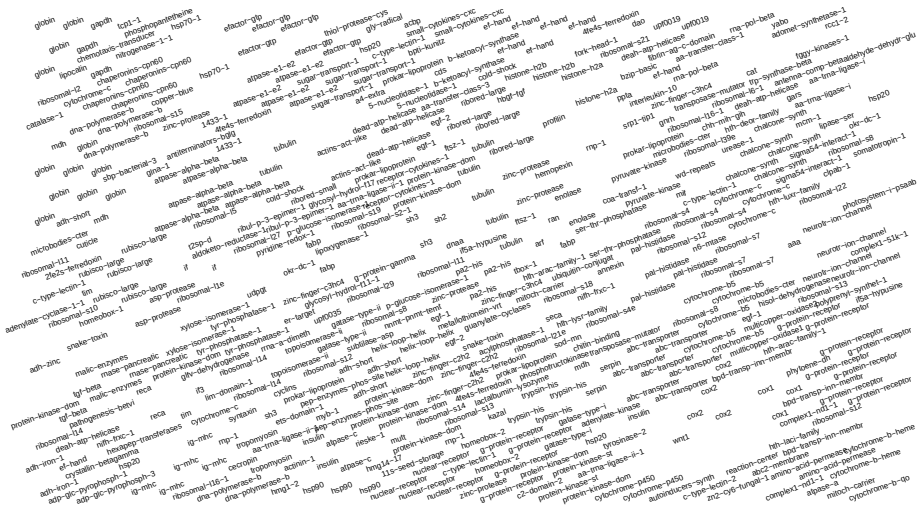

Fig. 3. Clustering of 77,977 protein sequences using a 30-by-20 unit SOM. The prototype sequences of the map nodes are the same as in Fig. 2. Each node is labeled by majority voting of the sequences having that node as their best-matching unit. The labels are the PROSITE classes [10] of the sequences.

and efactor-gtp there is a cluster of the hsp70 family. Tubulins are mapped to two closely located areas, one of which is characterized by alpha subunits and another by beta subunits, respectively.

Since there are altogether 1,352 classes in the PROSITE database, not all of them can be discussed in detail. But a general idea of the capability of the SOM can be gained by investigating the projections of the most frequent classes. Therefore the PROSITE classes were sorted according to their frequency in the SWISS-PROT database. The 32 most frequent classes were then projected on the SOM by finding the best-matching unit of each sequence belonging to the given class. The resulting class distributions are shown in Fig. 4.

In the visualization of the class distributions, some PROSITE classes were combined. For example, the actins class consists of 249 sequences of the family actins_act_like, 232 sequences of actins_2, and 227 sequences of actins_1. Trypsin_ser and trypsin_his were combined to the single trypsin class. Thiol_protease_asn, thiol_protease_his, and thiol_protease_ser were combined to the single thiol_protease class. Cytochrome_b class in the figure consists of both cytochrome_b_qo and cytochrome_b_heme. The distribution of protein_kinase_atp (1040 sequences) is not shown, because it was identical with the distribution of the protein_kinase_dom (1093 sequences).

Analyzing the cluster contents according to known protein families can give information about the specificity of the prototype sequences, like in the organization of the database performed on the basis of the sequence similarities. The

classification of the sequences according to the PROSITE classes, however, may also include structural information about the protein molecules. At any rate, many PROSITE classes were mapped to small and sharp areas on the SOM display.

Once the SOM has been trained, it is very fast to compute the projection of any new sequence. This requires only as many sequence comparisons as there are prototype sequences on the map. In the current work, the SOM contained 600 prototype sequences. Thus the work needed for classifying the new sequence into a prototype class is considerably lighter than comparison with all the 77,977 sequences of the whole database.

4 Discussion

This paper is based on the combination of two new possibilities: accessibility to masses of biological data in the Internet, and recent development of a clustering and visualization method that can cope with the masses of raw nonvectorial data in an unsupervised way.

The currently existing search engines for biological databases may give thousands of matches as a result of a short DNA sequence as a query sequence. The SOM can serve as a global visualization display, onto which also the results obtained by other means can be mapped. The sequence similarities can then be investigated based on the projections of the sequences on the map. The results for one query sequence can be all mutually similar or they can form distinct clusters, which can be reached by visual browsing.

The special Self-Organizing Map for symbolic items has been applied in this work for the first time to a major problem, self-organization of the 77,977 protein sequences of the SWISS-PROT database. Contrasted with earlier works, this extension of the SOM allows the use of any similarity measure for sequences. The resulting clustering and ordering of the data reflects the properties of the chosen similarity measure. The present result, where the similarities are computed by the FASTA method, is a two-dimensional map where similar proteins are mapped to the same node or neighboring nodes, and the structures of the clusters are thereby visualized, too. The geometrically organized picture makes it possible to illustrate the relationships of a large amount of sequences at a glance.

Since the SOM provides an ordered display of the representative prototype items of the data set, it may be used, e.g., for designing oligonucleotide or cDNA arrays (see [5] for a collection of reviews on microarray analysis). If the arrays were ordered using the SOM, similar oligonucleotides would be located close to each other in the array thus helping the visual interpretation of the data.

A great advantage of the SOM is that the basic form of the algorithm is very simple and straightforward to implement. It is therefore easy to apply the SOM to various tasks. The SOM can be used as a data mining and visualization tool for any data set, for which a similarity or distance measure between its elements can be defined.

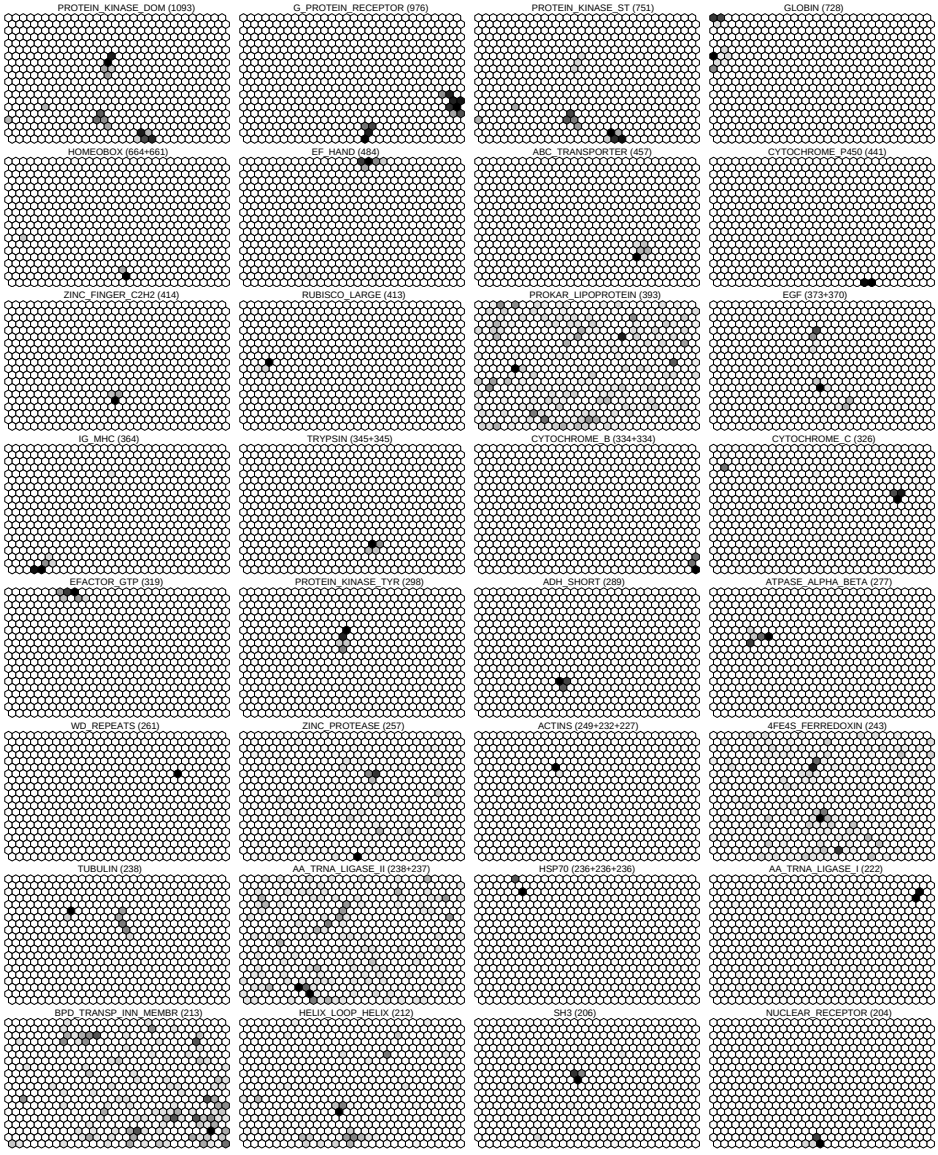

Fig. 4. Projections of the 32 most frequent PROSITE classes of the SWISS-PROT database on the SOM. Each subfigure represents the distribution of one class. The prototype sequences of the map nodes are the same as in Fig. 2. The shades of gray indicate the number of the protein sequences belonging to the given class in each map node. The maximum value (darkest shade of gray) is scaled to unity in each subfigure. The total number of the sequences in each class is shown in the parentheses after the PROSITE name.

References

1. Altschul, F., Gish, W., Miller, W., Myers, E., Lipman, D.: Basic local alignment search tool. J. Mol. Biol. **215** (1990) 403–410
2. Andrade, M., Casari, G., Sander, C., Valencia, A.: Classification of protein families and detection of the determinant residues with an improved self-organizing map. Biol. Cybern. **76** (1997) 441–450
3. Bairoch, A., Apweiler, R.: The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1999. Nucleic Acids Res. **27** (1999) 49–54
4. Cheng, Y.: Convergence and ordering of Kohonen's batch map. Neural Computation **9** (1997) 1667–1676
5. The chipping forecast. Nat. Genet. **21** (suppl.) (1999)
6. Ferrán, E., Ferrara, P.: Topological maps of protein sequences. Biol. Cybern. **65** (1991) 451–458
7. Ferrán, E., Pflugfelder, B., Ferrara, P.: Self-organized neural maps of human protein sequences. Protein Sci. **3** (1994) 507–521
8. Gray, R.: Vector quantization. IEEE ASSP Magazine **1**(2) (1984) 4–29
9. Hanke, J., Reich, J.: Kohonen map as a visualization tool for the analysis of protein sequences: multiple alignments, domains and segments of secondary structures. Comput. Appl. Biosci. **12**(6) (1996) 447–454
10. Hofmann, K., Bucher, P., Falquet, L., Bairoch, A.: The PROSITE database, its status in 1999. Nucleic Acids Res. **27** (1999) 215–219
11. Kohonen, T.: Self-organized formation of topologically correct feature maps. Biol. Cybern. **43** (1982) 59–63
12. Kohonen, T.: Median strings. Patt. Rec. Lett. **3** (1985) 309–313
13. Kohonen, T.: The self-organizing map. Proc. IEEE **78** (1990) 1464–1480
14. Kohonen, T.: Self-Organizing Maps. Springer-Verlag, Berlin-Heidelberg-New York (1995) (2nd ed. 1997)
15. Kohonen, T.: Self-organizing maps of symbol strings. Technical Report A42, Helsinki University of Technology, Laboratory of Computer and Information Science (1996)
16. Kohonen, T., Oja, E., Simula, O., Visa, A., Kangas, J.: Engineering applications of the self-organizing map. Proc. IEEE **84** (1996) 1358–1384
17. Kruskal, J., Wish, M.: Multidimensional scaling. Sage Publications, Newbury Park, CA (1978)
18. Pearson, W.: The FASTA program package. *ftp://ftp.virginia.edu/pub/fasta* (1999)
19. Pearson, W., Lipman, D.: Improved tools for biological sequence comparison. Proc. Natl. Acad. Sci. USA **85** (1988) 2444–2448
20. Smith, T., Waterman, M.: Comparison of biosequences. Adv. Appl. Math. **2** (1981) 483–489