Bioinformatics in neurocomputing framework

S.S. Ray, S. Bandyopadhyay, P. Mitra and S.K. Pal

Abstract: Different bioinformatics tasks like gene sequence analysis, gene finding, protein structure prediction and analysis, gene expression with microarray analysis and gene regulatory network analysis are described along with some classical approaches. The relevance of intelligent systems and neural networks to these problems is mentioned. Different neural network based algorithms to address the aforesaid tasks are then presented. Finally some limitations of the current research activity are provided. An extensive bibliography is included.

1 Introduction

Over the past few decades, major advances in the field of molecular biology, coupled with advances in genomic technologies, have led to an explosive growth in the biological information generated by the scientific community. This deluge of genomic information has, in turn, led to an absolute requirement for computerised databases to store, organise and index the data, and for specialised tools to view and analyse the data.

Bioinformatics can be viewed as 'the use of computational methods to make biological discoveries' [1]. It is an interdisciplinary field involving biology, computer science, mathematics and statistics to analyse biological sequence data, genome content and arrangement, and to predict the function and structure of macromolecules. The ultimate goal of the field is to enable the discovery of new biological insights in addition to create a global perspective from which unifying principles in biology can be derived [2]. There are three important sub-disciplines within bioinformatics:

a) development of new algorithms and models to assess different relationships among the members of a large biological data set in a way that allows researchers to access existing information and to submit new information as they are produced;

b) analysis and interpretation of various types of data including nucleotide and amino acid sequences, protein domains, and protein structures; and

c) development and implementation of tools that enable efficient access and management of different types of information.

Artificial neural networks (ANN), a biologically inspired technology, are machinery for adaptation and curve fitting and are guided by the principles of biological neural networks. ANN have been studied for many years with the hope of achieving human like performance, particularly in the field of pattern recognition. They are efficient adaptive and robust classifiers, producing near optimal solutions and achieving high speed via massive parallelism. Therefore, the application of ANN for solving certain problems in bioinformatics, which need optimisation of computation requirements, and robust, fast and close approximate solutions, appears to be appropriate and natural. Moreover, the errors generated in experiments with bioinformatics data can be handled with the robust characteristics of ANN and minimised during the trainnig process. The problem of integrating ANN and bioinformatics constitutes a new research area.

This article provides a survey of the various neural network based techniques that have been developed over the past few years for different bioinformatics tasks.

2 Elements of bioinformatics

Deoxyribonucleic acid (DNA) and proteins are biological macromolecules built as long linear chains of chemical components. DNA strands consist of a large sequence of nucleotides, or bases. For example there are more than three billion bases in human DNA sequences. DNA plays a fundamental role in different biochemical processes of living organisms in two respects. First it contains the templates for the synthesis of proteins, which are essential molecules for any organism [3]. The second role in which DNA is essential to life is as a medium to transmit hereditary information (namely the building plans for proteins) from generation to generation.

The units of DNA are called nucleotides. One nucleotide consists of one nitrogen base, one sugar molecule (deoxyribose) and one phosphate. Four nitrogen bases are denoted by one of the letters A (adenine), C (cytosine), G (guanine) and T (thymine). A linear strand of DNA is paired to a complementary strand. The complementary property stems from the ability of the nucleotides to establish specific pairs (A–T and G–C). The pair of complementary strands then forms the double helix that was first suggested by Watson and Crick in 1953. Each strand therefore carries the entire information and the biochemical machinery guarantees that the information can be copied over and over again even when the 'original' molecule has long since vanished.

A gene is primarily made up of a sequence of triplets of the nucleotides (exons). Introns (non-coding sequence) may also be present within a gene. Not all portions of the DNA sequences are coding. A coding zone indicates that it is a

IEE Proceedings online no. 20045051

doi:10.1049/ip-cds:20045051

Paper first received 16th June 2004 and in final revised form 5th October 2004

The authors are with Machine Intelligence Unit, Indian Statistical Institute, Kolkata 700108, India

E-mail: shubhra_r@isical.ac.in

template for a protein. As an example, for the human genome only 3–5% of the portions are coding, i.e., they constitute the gene. There are sequences of nucleotides within the DNA that are spliced out progressively in the process of transcription and translation. In brief, the DNA consists of three types of non-coding sequences (shown schematically in Fig. 1).

1. Intergenic regions: regions between genes that are ignored during the process of transcription.
2. Intragenic regions (or introns): regions within the genes that are spliced out from the transcribed RNA to yield the building blocks of the genes, referred to as exons.
3. Pseudogenes: genes that are transcribed into the RNA and stay there, without being translated, owing to the action of a nucleotide sequence.

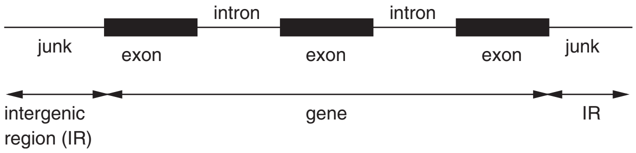


Fig. 1 *Various parts of DNA*

Proteins are made up of 20 different amino acids (or 'residues'), which are denoted by 20 different letters of the alphabet. Each of the 20 amino acids is coded by one or more triplets (or codons) of the nucleotides making up the DNA. Based on the genetic code the linear string of DNA is translated into a linear string of amino acids, i.e., a protein via mRNA [3].

3 Bioinformatics tasks

The different biological problems studied within the scope of bioinformatics can be broadly classified into two categories: genomics and proteomics which include genes, proteins, and amino acids. We describe below different tasks involved in their analysis along with their utilities.

3.1 Gene sequence analysis

The evolutionary basis of sequence alignment is based on the principles of similarity and homology [4]. Similarity is a quantitative measure of the fraction of two genes which are identical in terms of observable quantities. Homology is the conclusion drawn from data that two genes share a common evolutionary history; no metric is associated with this. The tasks of sequence analysis are as follows.

3.1.1 Sequence alignment: An alignment is a mutual arrangement of two or more sequences, that exhibits where the sequences are similar, and where they differ. An optimal alignment is one that exhibits the most correspondences and the least differences. It is the alignment with the highest score but may or may not be biologically meaningful. Basically there are two types of alignment methods, global alignment and local alignment. Global alignment [5] maximises the number of matches between the sequences along the entire length of the sequence. Local alignment [6] gives a highest scoring to local match between two sequences.

3.1.2 Pattern searching: This deals with searches for a nucleic pattern in a nucleic acid sequence, in a set of sequences or in a databank (e.g. INFO-BIOGEN) [7]. It is the potential for uncovering evolutionary relationships and patterns between different forms of life. With the aid of nucleotide and protein sequences, it should be possible to find the ancestral ties between different organisms. So far, experience indicates that closely related organisms have similar sequences and that more distantly related organisms have more dissimilar sequences. Proteins that show a significant sequence conservation indicating a clear evolutionary relationship are said to be from the same protein family. By studying protein folds (distinct protein building blocks) and families, scientists are able to reconstruct the evolutionary relationship between two species and to estimate the time of divergence between two organisms since they last shared a common ancestor.

3.1.3 Gene finding and promoter identification: In general a DNA strand consists of a large sequence of nucleotides, or bases. Owing to the huge size of the database, manual searching of genes, which code for proteins, is not practical. Therefore automatic identification of the genes from the large DNA sequences is an important problem in bioinformatics [8]. A cell mechanism recognises the beginning of a gene or gene cluster with the help of a promoter. The promoter is a region before each gene in the DNA that serves as an indication to the cellular mechanism that a gene is ahead. For example, the codon AUG (which codes for methionine) also signals the start of a gene. Recognition of regulatory sites in DNA fragments has become particularly popular because of the increasing number of completely sequenced genomes and mass application of DNA chips.

Promoters are key regulatory sequences that are necessary for the initiation of transcription. Experimental analysis has identified fewer than 10% of the potential promoter regions, assuming that there are at least 30,000 promoters in the human genome, one for each gene. On a genome-wide scale, pattern-based and genomic context-based computational approaches can suggest possible transcription factor-binding regions, but the rate of false-positive predictions is very high.

3.2 Protein analysis

Proteins are polypeptides, formed within cells as a linear chain of amino acids [9]. Within and outside of cells, proteins serve a myriad of functions, including structural roles (cytoskeleton), as catalysts (enzymes), transporters to ferry ions and molecules across membranes, and hormones to name just a few. There are twenty different amino acids that make up essentially all proteins on earth. Different tasks involved in protein analysis are as follows.

3.2.1 Multiple sequence alignment: Multiple amino acid sequence alignment techniques [1] are usually performed to fit one of the following scopes: (a) finding the consensus sequence of several aligned sequences; (b) helping in the prediction of the secondary and tertiary structures of new sequences; and (c) providing a preliminary step in molecular evolution analysis using phylogenetic methods for constructing phylogenetic trees.

In order to characterise protein families, one needs to identify shared regions of homology in a multiple sequence alignment; (this happens generally when a sequence search reveals homologies in several sequences). The clustering method can do alignments automatically but is subjected to some restrictions. Manual and eye validations are necessary in some difficult cases. The most practical and widely used method in multiple sequence alignment is the hierarchical extensions of pairwise alignment methods, where the principal is that multiple alignments are achieved by successive applications of pairwise methods.

3.2.2 Protein motif search:

A protein motif search [8] allows searching for a personal protein pattern in a sequence (personal sequence or an entry in a gene bank). Proteins are derived from a limited number of basic building blocks (domains). Evolution has shuffled these modules giving rise to a diverse repertoire of protein sequences, as a result proteins can share a global or local relationship. Protein motif search is a technique for searching sequence databases to uncover common domains/motifs of biological significance that categorise a protein into a family.

3.2.3 Structural genomics:

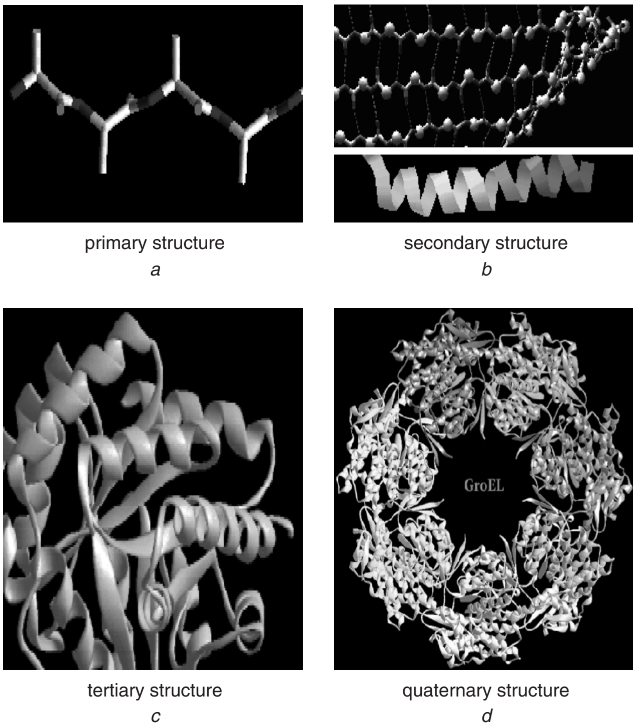
Structural genomics is the prediction of the 3-dimensional structure of a protein from the primary amino acid sequence [10]. This is one of the most challenging tasks in bioinformatics. The four levels of protein structure (Fig. 2) are

(a) primary structure: the sequence of amino acids that compose the protein,

(b) secondary structure: the spatial arrangement of the atoms constituting the main protein backbone, such as alpha helices and beta strands,

(c) tertiary structure: formed by packing secondary structural elements into one or several compact globular units called domains, and

(d) final protein may contain several polypeptide chains arranged in a quaternary structure.



primary structure
a

secondary structure
b

tertiary structure
c

quaternary structure
d

Fig. 2 *Different levels of protein structures*

Sequence similarity methods can predict the secondary and tertiary structures based on homology to known proteins. Secondary structure prediction can be made using Chou–Fasman [10], GOR, neural network, and nearest neighbour methods. Methods for tertiary structure prediction involve energy minimisation, molecular dynamics, and stochastic searches of conformational space.

3.3 Gene expression and microarrays

Gene expression is the process by which a gene's coded information is converted into the structures present and operating in the cell. Expressed genes include those that are transcribed into mRNA and then translated into protein and those that are transcribed into RNA but not translated into protein (e.g., transfer and ribosomal RNA). Not all genes are expressed and gene expression involves the study of the expression level of genes in the cells under different conditions. Conventional wisdom is that gene products that interact with each other are more likely to have similar expression profiles than if they do not [11].

Microarray technology [12] allows expression levels of thousands of genes to be measured at the same time. Comparison of gene expression between normal and diseased (e.g., cancerous) cells are also done by microarray. There are several names for this technology for example DNA microarrays, DNA arrays, DNA chips, gene chips. A microarray is typically a glass (or some other material) slide, on to which DNA molecules are attached at fixed locations (spots). There may be tens of thousands of spots on an array, each containing a huge number of identical DNA molecules (or fragments of identical molecules), of lengths from twenty to hundreds of nucleotides. For gene expression studies, each of these molecules ideally should identify one gene or one exon in the genome, however, in practice this is not always so simple and may not even be generally possible owing to families of similar genes in a genome. The spots are either printed on the microarrays by a robot, or synthesised by photolithography (similar to computer chip production) or by ink-jet printing.

Many unanswered, and important, questions could potentially be answered by correctly selecting, assembling, analysing, and interpreting microarray data. Clustering is commonly used in microarray experiments to identify groups of genes that share similar expressions. Genes that are similarly expressed are often co-regulated and involved in the same cellular processes. Therefore, clustering suggests functional relationships between groups of genes. It may also help in identifying promoter sequence elements that are shared among genes. In addition, clustering can be used to analyse the effects of specific changes in experimental conditions and may reveal the full cellular responses triggered by those conditions.

3.4 Gene regulatory network analysis

Another important and interesting question in biology is how gene expression is switched on and off, i.e., how genes are regulated [1]. Since almost all cells in a particular organism have an identical genome, differences in gene expression and not the genome content are responsible for cell differentiation (how different cell types develop from a fertilised egg) during the life of the organism.

Gene regulation in eukaryotes, is not well understood, but there is evidence that an important role is played by a type of proteins called transcription factors. The transcription factors can attach (bind) to specific parts of the DNA, called transcription factor binding sites (i.e., specific, relatively short combinations of A, T, C or G), which are located in so-called promoter regions. Specific promoters are associated with particular genes and are generally not too far from the respective genes, though some regulatory effects can be located as far as 30,000 bases away, which makes the definition of the promoter difficult.

Transcription factors control gene expression by binding the gene's promoter and either activating (switching on) the gene's transcription, or repressing it (switching it off). Transcription factors are gene products themselves, and therefore, in turn, can be controlled by other transcription factors. Transcription factors can control many genes, and some (probably most) genes are controlled by combinations of transcription factors. Feedback loops are possible. Therefore we can talk about gene regulation networks. The understanding, describing and modelling of such gene regulation networks is one of the most challenging problems in functional genomics. Microarrays and computational methods are playing a major role in attempts to reverse engineer gene networks from various observations. Note that in reality the gene regulation is likely to be a stochastic and not a deterministic process. Traditionally molecular biology has followed a so-called reductionist approach mostly concentrating on a study of a single or very few genes in any particular research project. With genomes being sequenced, this is now changing into a so-called systems approach.

4 Relevance of neural networks in bioinformatics

Artificial neural network (ANN) models try to emulate the biological neural network with electronic circuitry. Recently, ANNs have found widespread use for classification tasks and function approximation in many fields of medicinal chemistry and bioinformatics. For these kinds of data analysis mainly two types of networks are employed; the ‘supervised’ neural network (SNN) and the ‘unsupervised’ neural network (UNN). The main applications of SNNs (e.g. multilayer perceptrons (MLPs) are feedforward neural networks trained with the standard backpropagation algorithm) are function approximation, classification, pattern recognition and feature extraction, and prediction. Moreover, they are able to detect second and higher order correlations in patterns. This is specially important in biological systems, which frequently display nonlinear behaviour. These networks require a set of molecular compounds with known activities to model structure-activity relationships and are able to determine the relevant features in the data set, usually by means of training processes. This principle coined the term ‘supervised’ networks. Correspondingly, ‘unsupervised’ networks (e.g. Kohonen self-organising maps) can be applied to clustering and feature extraction tasks even without prior knowledge of molecular activities or properties. Unsupervised learning has the advantage that no previous knowledge about the system under study is required.

The main characteristics of ANNs are:

a) adaptability to new data/environment,

b) robustness/ruggedness to failure of components,

c) speed via massive parallelism, and

d) optimality w.r.t. error.

Let us now explain the functioning of an ANN in bioinformatics with an example of protein secondary structure prediction from a linear sequence of amino acids (Fig. 3).

Step 1: In the ANN usually a certain number of input ‘nodes’ are each connected to every node in a hidden layer.

Step 2: Every residue in a protein data bank (PDB) entry can be associated to one of the three secondary structures (helix, sheet or neither: coil). ANNs are designed with 21 input nodes (one for each residue including a null residue) and three output nodes coding for each of the three possible secondary structure assignments (helix, sheet and coil).

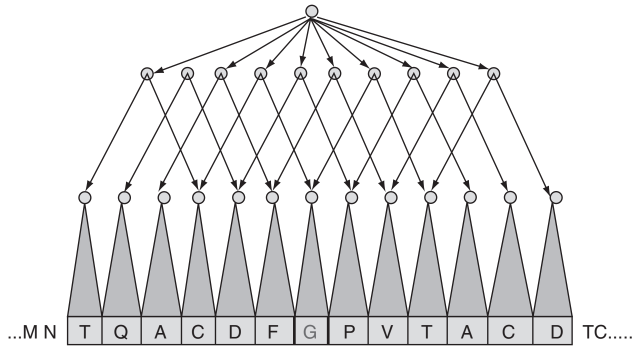


Fig. 3 *A linear chain of amino acids is applied as input to the ANN*

Step 3: Each node in the hidden layer is then connected to every node in the final output layer.

Step 4: The input and output nodes are restricted to binary values (1 or 0) when loading the data onto the network during training and the weights are then modified by the program itself during the training process.

Step 5: Helix can be coded as 0, 0, 1 on the three output nodes; sheet can be coded as 0, 1, 0 and coil as 1, 0, 0. A similar binary coding scheme can be used for the 20 input nodes for the 20 amino acids.

Step 6: To consider a moving window of n residues at a time, input layer should contain $20 \times n$ nodes plus one node at each position for a null residue.

Step 7: Each node will ‘decide’ to send a signal to the nodes it is connected to, based on evaluating its transfer function after all of its inputs and connection weights have been summed.

Step 8: Over 100 protein structures were used to train the network.

Step 9: Training proceeds by holding a particular data constant onto both the input and output nodes and iterating the network in a process that modifies the connection weights until the changes made to them approach zero.

Step 10: When such convergence is reached, the network is said to be trained and is ready to receive new (unknown) experimental data.

Step 11: Now the connection weights are not changed and the values of the hidden and output nodes are calculated in order to determine the structure of the input sequence of proteins.

Selection of unbiased and normalised training data, however, is probably just as important as the network architecture in the design of a successful NN.

5 Anns in bioinformatics

Let us now describe the different attempts made using ANNs in certain tasks of bioinformatics in the broad domains of sequence analysis, structure prediction, and gene analysis described in Section 3.

5.1 Sequence alignment

Given inputs extracted from an aligned column of DNA bases and the underlying Perkin Elmer Applied Biosystems (ABI) fluorescent traces, Allex *et al.* [13] trained a neural network to determine correctly the consensus base for the column. They compared five representations empirically; one uses only base calls and the others include trace information. The networks that incorporate trace information into their input representations attained the most accurate results for consensus sequence. Consensus accuracies ranging from 99.26% to 99.98% are acheived for coverages from two to six aligned sequences. In contrast, the network that only uses base calls in its input representation has over double that error rate.

In [14] a molecular alignment method with the Hopfield neural network (HNN) is discussed. Molecules are represented by four kinds of chemical properties (hydrophobic group, hydrogen-bonding acceptor, hydrogen-bonding donor, and hydrogen-bonding donor/acceptor), and then those properties between two molecules correspond to each other using HNN. The method is applied to three-dimensional quantitative structure-activity relationship (3D-QSAR) analysis and it reproduced successfully the real molecular alignments obtained from X-ray crystallography.

GenTHREADER is a neural network architecture that predicts similarity between gene sequences [15]. The effects of sequence alignment score and pairwise potential are the network outputs. GenTHREADER was used successfully for the structure prediction in two cases: case 1: ORF MG276 from *Mycoplasma genitalium* was predicted to share structure similarity with 1HGX; case 2: MG276 shares a low sequence similarity (10% sequence identity) with 1HGX.

A back-propagation neural network can grossly approximate the score function of the popular BLAST family of genomic sequence alignment and scoring tools. The resultant neural network may provide a processing speed advantage over the BLAST tool, but may suffer somewhat in comparison to the accuracy of BLAST. Further study is necessary to determine whether a neural network with additional hidden units or structural complexity could be used to more closely approximate BLAST. However, closer approximation may also limit the speed performance advantages enjoyed by the neural network approach.

Other related investigations in sequence analysis are available in [16, 17].

5.2 Gene finding and promoter identification

The application of artificial neural networks for discriminating the coding system of eukaryotic genes is investigated in [18]. Over 300 genes from eight eukaryotic organisms are chosen: human, mouse, rat, horse, ox, sheep, soybean and rabbit. From these genes different discrimination models are build which are relevant to genes promoter regions, poly(A) signals, splice site locations of introns and noose structures. The results showed that as long as the coding length is definite, the only correct coding region can be chosen from the large number of possible solutions discriminated by neural networks.

In [19] the quantitative similarity among tRNA gene sequences was acquired by analysis with an artificial neural network. The evolutionary relationship derived from ANN results was consistent with those from other methods. A new sequence was recognised to be a tRNA-like gene by a neural network on the analysis of similarity.

The work of Lukashin *et al.* [20] is one of the earlier investigations that discussed the problem of recognition of promoter sites in the DNA sequence in a neural network framework. The learning process involves a small (of the order of 10%) part of the total set of promoter sequences. During this procedure the neural network develops a system of distinctive features (key words) to be used as a reference in identifying promoters against the background of random sequences. The learning quality is then tested with the whole set. The efficiency of promoter recognition has been reported as 94 to 99% and the probability of an arbitrary sequence being identified as a promoter is 2 to 6%.

In [21] a multilayered feed-forward ANN architecture is trained for predicting whether a given nucleotide sequence is a mycobacterial promoter sequence. The ANN is used in conjunction with the caliper randomisation (CR) approach for determining the structurally/functionally important regions in the promoter sequences. This work shows that ANNs are efficient tools for predicting mycobacterial promoter sequences and determining structurally/functionally important sub-regions therein.

Other related investigations in promoter identification are available in [22, 23].

5.3 Protein analysis

The most successful techniques for prediction of the three-dimensional structure of protein rely on aligning the sequence of a protein of unknown structure to a homologue of known structure. Such methods fail if there is no homologue in the structural database, or if the technique for searching the structural database is unable to identify homologues that are present.

The work of Qian *et al.* [24] is one of the earlier investigations that discussed the protein structure prediction problem in a neural network framework. They used X-ray-derived crystal structures of globular proteins available at that time to train a NN to predict the secondary structure of non-homologous proteins. Over 100 protein structures were used to train this network. After training, when the NN was queried with new data, a prediction accuracy of 64% was obtained.

Rost *et al.* [25, 26] took advantage of the fact that a multiple sequence alignment contains more information about a protein than the primary sequence alone. Instead of using a single sequence as input into the network, they used a sequence profile that resulted from the multiple alignments. This resulted in a significant improvement in prediction accuracy to 71.4%. Recently, more radical changes to the design of NNs including bi-directional training and the use of the entire protein sequence as simultaneous input instead of a shifting window of fixed length has led to prediction accuracy above 71%.

The prediction of protein secondary structure using structured neural networks and multiple sequence alignments have been investigated by Riis and Krogh [27]. Separate networks are used for predicting the three secondary structures, ff-helix, fi-strand and coil. The networks are designed using *a priori* knowledge of amino acid properties with respect to the secondary structure and of the characteristic periodicity in ff-helices. This method gives an overall prediction accuracy of 66.3% when using seven-fold cross-validation on a database of 126 non-homologous globular proteins. Applying the method to multiple sequence alignments of homologous proteins increases the prediction accuracy significantly to 71.3% [27].

In [28] a method has been developed using ANNs for the prediction of beta-turn types I, II, IV and VIII. For each turn type, two consecutive feed-forward back-propagation networks with a single hidden layer have been used. The first sequence-to-structure network has been trained on single sequences in addition to on PSI-BLAST PSSM. The output from the first network along with PSIPRED [29] predicted secondary structure has been used as input for the second-level structure-to-structure network. The networks have been trained and tested on a non-homologous data set of 426 proteins chains by seven-fold cross-validation. The prediction performance for each turn type is improved by using multiple sequence alignment, second level structure-to-structure network and PSIPRED predicted secondary structure information.

The back-propagation neural network algorithm is a commonly used method for predicting the secondary structure of proteins. Wood *et al.* [30] compared the cascade-correlation ANN architecture [31] with the back-propagation ANN using a constructive algorithm and found that cascade-correlation achieves predictive accuracies comparable to those obtained by back-propagation, in shorter time. Ding *et al.* [32] used support vector machine (SVM) and the neural network (NN) learning methods as base classifiers for protein fold recognition, without relying on sequence similarity.

Other related investigations in protein structure prediction are available in [33–38].

5.4 Gene expression and microarray

Clustering is commonly used in microarray experiments to identify groups of genes that share similar expression. Genes that are similarly expressed are often co-regulated and involved in the same cellular processes. Therefore, clustering suggests functional relationships between groups of genes. It may also help in identifying promoter sequence elements that are shared among genes. In addition, clustering can be used to analyse the effects of specific changes in experimental conditions and may reveal the full cellular responses triggered by those conditions.

Most of the analysis of the enormous amount of information provided on microarray chips with regard to cancer patient prognosis has relied on clustering techniques and other standard statistical procedures. These methods are inadequate in providing the reduced gene subsets required for perfect classification. ANNs trained on microarray data from DLBCL lymphoma patients have, for the first time, been able to predict the long-term survival of individual patients with 100% accuracy [39]. Here it is shown that differentiating the trained network can narrow the gene profile to less than three dozen genes for each classification and artificial neural networks are superior tools for digesting microarray data.

Sawa *et al.* [40] described a neural network-based similarity index as a nonlinear similarity index and compared the results with other proximity measures for *Saccharomyces cerevisiae* gene expression data. Here it is shown that the clusters obtained using Euclidean distance, correlation coefficients, and mutual information were not significantly different. The clusters formed with the neural network-based index were more in agreement with those defined by functional categories and common regulatory motifs.

Diffuse large B-cell lymphoma (DLBCL) is the largest category of aggressive lymphomas. Less than 50% of patients can be cured by combination chemotherapy. Microarray technologies have recently shown that the response to chemotherapy reflects the molecular heterogeneity in DLBCL. On the basis of published microarray data, Ando *et al.* [41] described a fuzzy neural network (FNN) model to analyse gene expression profiling data for the precise and simple prediction of survival of DLBCL patients. From data on 5857 genes, this model identified four genes (CD10, AA807551, AA805611 and IRF-4) that could be used to predict prognosis with 93% accuracy. FNNs are powerful tools for extracting significant biological markers affecting prognosis, and are applicable to various kinds of expression profiling data for any malignancy.

Bicciato *et al.* [42] described a computational procedure for pattern identification, feature extraction, and classification of gene expression data through the analysis of an autoassociative neural network model. The identified patterns and features contain critical information about gene-phenotype relationships observed during changes in cell physiology. The methodology has been tested on two different microarray datasets, acute human leukemia and the human colon adenocarcinoma.

The Bayesian neural network is used with structural learning with forgetting for searching optimal network size and structure of microarray data in order to capture the structural information of gene expressions [43]. The process of Bayesian learning starts with a feed forward neural network (FFNN) and prior distribution for the network parameters. The prior distribution gives initial beliefs about the parameters before any data is observed. After new data are observed, the prior distribution is updated to the posterior distribution using Bayes rules. Multi-layer perceptron (MLP) is mainly considered as the network structure for Bayesian learning. Since the correlated data may include high levels of noise, efficient regularisation techniques are required to improve the generalisation performance. This involves network complexity adjustment and performance function modification. To do the latter, instead of the sum of squared error (SSE) on the training set, a cost function is automatically adjusted.

Vohradsky [44] used artificial neural networks as models of the dynamics of gene expression. The significance of the regulatory effect of one gene product on the expression of other genes of the system is defined by a weight matrix. The model considers multigenic regulation including positive and/or negative feedback. The process of gene expression is described by a single network and by two linked networks where transcription and translation are modelled independently. Each of these processes is described by different networks controlled by different weight matrices. Methods for computing the parameters of the model from experimental data are also shown.

Plausible neural network (PLANN) is another universal data analysis tool based upon artificial neural networks and is capable of plausible inference and incremental learning [45]. This tool has been applied to research data from molecular biological systems through the simultaneous analysis of gene expression data and other types of biological information.

Relevant investigations for gene expression and microarray are also available in [46].

5.5 Gene regulatory network

Adaptive double self-organising map (ADSOM) [47] provides a novel clustering technique for identifying gene regulatory networks. It has a flexible topology and it performs clustering and cluster visualisation simultaneously, thereby requiring no *a priori* knowledge about the number of clusters. ADSOM is developed based on a recently introduced technique known as double self-organising map (DSOM). DSOM combines features of the popular self-organising map (SOM) with two-dimensional position vectors, which serve as a visualisation tool to decide how many clusters are needed. Although DSOM addresses the problem of identifying unknown number of clusters, its free parameters are difficult to control to guarantee correct results and convergence. ADSOM updates its free parameters during training and it allows convergence of its position vectors to a fairly consistent number of clusters provided that its initial number of nodes is greater than the expected number of clusters. The number of clusters can be identified by visually counting the clusters formed by the position vectors after training. The reliance of ADSOM in identifying the number of clusters is proven by applying it to publicly available gene expression data from multiple biological systems such as yeast, human, and mouse. It may be noted that gene regulatory network analysis is a very recent research area, and neural network applications to it are scarce.

Appropriate definition of neural network architecture prior to data analysis is crucial for successful data mining. This can be challenging when the underlying model of the data is unknown. Using simulated data, Ritchie *et al.* [48] optimised back-propagation neural network architecture using genetic programming to improve the ability of neural networks to model, identify, characterise and detect nonlinear gene-gene interactions in studies of common human diseases. They showed that the genetic programming optimised neural network is superior to the traditional back-propagation neural network approach in terms of predictive ability and power to detect gene-gene interactions when non-functional polymorphisms are present.

6 Other bioinformatics tasks using ANNs

Dopazo *et al.* [49] described a new type of unsupervised growing self-organising neural network that expands itself following the taxonomic relationships existing among the sequences being classified. The binary tree topology of this neural network, opposite to other more classical neural network topologies, permits an efficient classification of sequences. The growing nature of this procedure allows to stop it at the desired taxonomic level without the necessity of waiting until a complete phylogenetic tree is produced. The time for convergence is approximately a linear function of the number of sequences. This neural network methodology is an excellent tool for the phylogenetic analysis of a large number of sequences.

Parbhane *et al.* [50] utilise an artificial neural network (ANN) for the prediction of DNA curvature in terms of retardation anomaly. The ANN captured the phase information and increased helix flexibility. Base pair effects in determining the extent of DNA curvature has been developed. The network predictions validate the known experimental results and also explain how the base pairs affect the curvature. The results suggest that ANN can be used as a model-free tool for studying DNA curvature.

Drug resistance is a very important factor influencing the failure of current HIV therapies. The ability to predict the drug resistance of HIV protease mutants may be useful in developing more effective and longer lasting treatment regimens. The HIV resistance is predicted to two current protease inhibitors, Indinavir and Saquinavir. This problem is handled in [51] from two perspectives. First, a predictor was constructed based on the structural features of the HIV protease-drug inhibitor complex. A particular structure was represented by its list of contacts between the inhibitor and the protease. Next, a classifier was constructed based on the sequence data of various drug resistant mutants. In both cases, SOMs were first used to extract the important features and cluster the patterns in an unsupervised manner. This was followed by subsequent labelling based on the known patterns in the training set. The classifier using the structure information is able to correctly recognise the previously unseen mutants with an accuracy of between 60 and 70%. The method is superior to a random classifier.

In [52] an ANN is trained to predict the sequence of the human TP53 tumor suppressor gene based on a p53 GeneChip. The trained neural network uses as input the fluorescence intensities of DNA hybridised to oligonucleotides on the surface of the chip. In this methodology errors are reported between zero and four in the predicted 1300 bp sequence when tested on wild-type TP53 sequence.

Neural network computations on DNA and RNA sequences are used in [53] to demonstrate that data compression is possible in these sequences. The result implies that a certain discrimination should be achievable between structured and random regions. The technique is illustrated by computing the compressibility of short RNA sequences such as tRNA.

A basic description of artificial neural networks and applications of neural nets to problems in human gene finding for three different types of data are discussed in [54].

7 Conclusion and scope of future research

Artificial neural networks (ANNs) are the first group of machine learning algorithms to be used on a biological pattern recognition problem. The rationale for applying computational approaches to facilitate the understanding of various biological processes are mainly:

- To provide a more global perspective in experimental design.
- To capitalise on the emerging technology of database-mining – the process by which testable hypotheses are generated regarding the function or structure of a gene or protein of interest by identifying similar sequences in better characterised organisms.

Neural networks appear to be a very powerful artificial intelligence (AI) paradigm to handle these issues [55]. The most important, and attractive, feature of ANNs is their capability of learning (generalising) from example (extracting knowledge from data). This feature makes the ANN an attractive choice for bioinformatics tasks. The combination of backpropagation learning algorithm and the feed-forward, layered networks have been applied to virtually all pattern recognition problems (like sequence analysis, protein analysis, gene finding) in bioinformatics. The reason for this is the simplicity of the algorithm, and the vast body of research that has studied these networks. Although these networks are theoretically capable of separating a problem space into appropriate classes irrespective of the complexity of the separation boundaries, one of the classical disadvantages of these networks is that a certain amount of a priori knowledge is required in order to build a useful network. A crucial factor in training a useful network is its size (number of layers, size of layers, and number of synaptic connections). In many cases, it takes a large number of simulations before a close-to-optimum size of the network is found. If the network is designed to be larger than this optimum size, it will memorise (also called over-fit) the data rather than generalising and extracting knowledge. If the network is chosen to be smaller than the optimum size, the network will never learn the entire task at hand. However, there have been several reports dealing with the determination of an appropriate size of a network for a particular task.

Let us consider self-organising map (SOM), as an example, which has been widely used in mining biological data. SOM has the distinct advantage that they allow a priori knowledge to be included in the clustering process and well suited for analysing patterns (e.g., microarray data). They are ideally suited to exploratory data analysis, allowing one to impose partial structure on the clusters (in contrast to the rigid structure of hierarchical clustering, the strong prior hypotheses used in Bayesian clustering, and the nonstructure of k-means clustering) facilitating easy visualisation and interpretation. SOMs have good computational properties and are easy to implement, reasonably fast, and are scalable to large data sets. The most prominent disadvantage of the SOM-based approach is that it is difficult to know when to stop the algorithm and it may get stuck to a local minima, so the map is allowed to grow indefinitely to a point where clearly different sets of patterns are identified.

Other soft computing tools, like fuzzy set theory and genetic algorithms, integrated with ANN [56] may also be used; based on the principles of case based reasoning [57]. Even though the current approaches in biocomputing are very helpful in identifying patterns and functions of proteins and genes, they are still far from being perfect. They are not only time-consuming, requiring Unix workstations to run on, but might also lead to false interpretations and assumptions due to necessary simplifications. It is therefore still mandatory to use biological reasoning and common sense in evaluating the results delivered by a biocomputing program. Also, for evaluation of the trustworthiness of the output of a program it is necessary to understand the mathematical/theoretical background of it to finally come up with a useful and senseful analysis.

8 Acknowledgment

This work is partly supported by grant 22(0346)/02/EMR-II of the Council of Scientific and Industrial Research (CSIR), New Delhi, under the project “Knowledge Based Connectionist Data Mining System: Design and Application”.

9 References

1 Baldi, P., and Brunak, S.: ‘Bioinformatics: the machine learning approach’ (MIT Press, Cambridge, MA, 1998)

2 Altman, R.B., Valencia, A., Miyano, S., and Ranganathan, S.: ‘Challenges for intelligent systems in biology’, *IEEE Intell. Syst.*, 2001, **16**, (6), pp. 14–20

3 Setubal, J., and Meidanis, J.: ‘Introduction to computational molecular biology’ (International Thomson Publishing, Boston, MA, 1999)

4 Nash, H., Blair, D., and Grefenstette, J.: ‘Comparing algorithms for large-scale sequence analysis’. Proc. 2nd IEEE Int. Symp. on Bioinformatics and Bioengineering (BIBE’01), 2001, pp. 89–96

5 Needleman, S.B., and Wunsch, C.D.: ‘A general method applicable to the search for similarities in the amino acid sequence of two proteins’, *J. Mol. Biol.*, 1970, **48**, pp. 443–453

6 Smith, T.F., and Waterman, M.S.: ‘Identification of common molecular sequences’, *J. Mol. Biol.*, 1981, **147**, pp. 195–197

7 Gautheret, D., Major, F., and Cedergren, R.: ‘Pattern searching/alignment with RNA primary and secondary structures: an effective descriptor for tRNA’, *Comp. Appl. Biosci.*, 1990, **6**, pp. 325–331

8 Fickett, J.W.: ‘Finding genes by computer: the state of the art’, *Trends Genet.*, 1996, **12**, (8), pp. 316–320

9 Salzberg, S.L., Searls, D.B., and Kasif, S.: ‘Computational methods in molecular biology’ (Elsevier Science, Amsterdam, 1998)

10 Chou, P., and Fasmann, G.: ‘Prediction of the secondary structure of proteins from their amino acid sequence’, *Adv. Enzym.*, 1978, **47**, pp. 145–148

11 Luscombe, N.M., Greenbaum, D., and Gerstein, M.: ‘What is bioinformatics? A proposed definition and overview of the field’, *Methods Informat. Med.*, 2001, **40**, (4), pp. 346–358

12 Quackenbush, J.: ‘Computational analysis of microarray data’, *Nat. Rev. Genetics*, 2001, **2**, pp. 418–427

13 Allex, C.F., Shavlik, J.W., and Blattner, F.R.: ‘Neural network input representations that produce accurate consensus sequences from DNA fragment assemblies’, *Bioinformatics*, 1999, **15**, (9), pp. 723–728

14 Arakawa, M., Hasegawa, K., and Funatsu, K.: ‘Application of the novel molecular alignment method using the Hopfield neural network to 3D-QSAR’, *J. Chem. Inf. Comput. Sci.*, 2003, **43**, (5), pp. 1396–1402

15 Jones, D.T.: ‘GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences’, *J. Mol. Biol.*, 1999, **287**, pp. 797–815

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

17 Petersen, S.B., Bohr, H., Bohr, J., Brunak, S., Cotterill, R.M., Fredholm, H., and Lautrup, B.: ‘Training neural networks to analyse biological sequences’, *Trends Biotechnol.*, 1990, **8**, (11), pp. 304–308

18 Cai, Y., and Chen, C.: ‘Artificial neural network method for discriminating coding regions of eukaryotic genes’, *Comput. Appl. Biosci.*, 1995, **11**, (5), pp. 497–501

19 Sun, J., Song, W.Y., Zhu, L.H., and Chen, R.S.: ‘Analysis of tRNA gene sequences by neural network’, *J. Comput. Biol.*, 1995, **2**, (3), pp. 409–416

20 Lukashin, A.V., Anshelevich, V.V., Amirikyan, B.R., Gragerov, A.I., and Frank-Kamenetskii, M.D.: ‘Neural network models for promoter recognition’, *J. Biomol. Struct. Dyn.*, 1989, **6**, (6), pp. 1123–1133

21 Kalate, R.N., Tambe, S.S., and Kulkarni, B.D.: ‘Artificial neural networks for prediction of mycobacterial promoter sequences’, *Comput. Biol. Chem.*, 2003, **27**, (6), pp. 555–564

22 Reese, M.G.: ‘Application of a time-delay neural network to promoter annotation in the *Drosophila melanogaster* genome’, *Comput. Chem.*, 2001, **26**, (1), pp. 51–56

23 Mahadevan, I., and Ghosh, I.: ‘Analysis of *E. coli* promoter structures using neural networks’, *Nucleic Acids Res.*, 1994, **22**, (11), pp. 2158–2165

24 Qian, N., and Sejnowski, T.J.: ‘Predicting the secondary structure of globular proteins using neural network models’, *J. Mol. Biol.*, 1988, **202**, (4), pp. 865–884

25 Rost, B., and Sander, C.: ‘Improved prediction of protein secondary structure by use of sequence profiles and neural networks’, *Proc. Nat. Acad. Sci.*, 1993, **90**, (16), pp. 7558–7562

26 Rost, B., and Sander, C.: ‘Prediction of protein secondary structure at better than 70% accuracy’, *J. Mol. Biol.*, 1993, **232**, pp. 584–599

27 Riis, S.K., and Krogh, A.: ‘Improving prediction of protein secondary structure using structured neural networks and multiple sequence alignments’, *J. Comput. Biol.*, 1996, **3**, pp. 163–183

28 Kaur, H., Raghava, G.P.: ‘A neural network method for prediction of beta-turn types in proteins using evolutionary information’, *Bioinformatics*, 2004, **20**, (16), pp. 2751–2758

29 McGuffin, L.J., Bryson, K., and Jones, D.T.: ‘The PSIPRED protein structure prediction server’, *Bioinformatics*, 2000, **16**, (4), pp. 404–405

30 Wood, M.J., and Hirst, J.D.: ‘Predicting protein secondary structure by cascade-correlation neural networks’, *Bioinformatics*, 2004, **20**, (3), pp. 419–420

31 Pasquier, C., Promponas, V.J., and Hamodrakas, S.J.: ‘PRED-CLASS: cascading neural networks for generalized protein classification and genome-wide applications’, *Proteins*, 2001, **44**, (3), pp. 361–369

32 Ding, C.H., and Dubchak, I.: ‘Multi-class protein fold recognition using support vector machines and neural networks’, *Bioinformatics*, 2001, **17**, (4), pp. 349–358

33 Berry, E.A., Dalby, A.R., and Yang, Z.R.: ‘Reduced bio basis function neural network for identification of protein phosphorylation sites: comparison with pattern recognition algorithms’, *Comput. Biol. Chem.*, 2004, **28**, (1), pp. 75–85

34 Shepherd, A.J., Gorse, D., and Thornton, J.M.: ‘A novel approach to the recognition of protein architecture from sequence using Fourier analysis and neural networks’, *Proteins*, 2003, **50**, (2), pp. 290–302

35 Pollastri, G., Baldi, P., Fariselli, P., and Casadio, R.: ‘Improved prediction of the number of residue contacts in proteins by recurrent neural networks’, *Bioinformatics*, 2001, **17**, (1), pp. 234–242

36 Lin, K., May, A.C., and Taylor, W.R.: ‘Threading using neural nEtwork (TUNE): the measure of protein sequence-structure compatibility’, *Bioinformatics*, 2002, **18**, (10), pp. 1350–1357

37 Cai, Y.D., Liu, X.J., and Chou, K.C.: ‘Prediction of protein secondary structure content by artificial neural network’, *J. Comput. Chem.*, 2003, **24**, (6), pp. 727–731

38 Dietmann, S., and Frommel, C.: ‘Prediction of 3D neighbours of molecular surface patches in proteins by artificial neural networks’, *Bioinformatics*, 2002, **18**, (1), pp. 167–174

39 O’Neill, M.C., and Song, L.: ‘Neural network analysis of lymphoma microarray data: prognosis and diagnosis near-perfect’, *BMC Bioinformatics*, 2003, **4**, (1), pp. 13–20

40 Sawa, T., and Ohno-Machado, L.: ‘A neural network-based similarity index for clustering DNA microarray data’, *Comput. Biol. Med.*, 2003, **33**, (1), pp. 1–15

41 Ando, T., Suguro, M., Hanai, T., Kobayashi, T., Honda, H., and Seto, M.: ‘Fuzzy neural network applied to gene expression profiling for predicting the prognosis of diffuse large B-cell lymphoma’, *Jpn. J. Cancer. Res.*, 2002, **93**, (11), pp. 1207–1212

42 Bicciato, S., Pandin, M., Didone, G., and Di Bello, C.: ‘Pattern identification and classification in gene expression data using an autoassociative neural network model’, *Biotechnol. Bioeng.*, 2003, **81**, (5), pp. 594–606

43 Liang, Y., Georgre, E.O., and Kelemen, A.: ‘Bayesian neural network for microarray data’. Technical Report, Department of Mathematical Sciences, University of Memphis, Memphis, TN 38152, USA

44 Vohradsky, J.: ‘Neural network model of gene expression’, *FASEB J.*, 2001, **15**, (3), pp. 846–854

45 PLANN Software, ‘PNN Technologies’, *Pasadena*, CA, USA

46 Herrero, J., Valencia, A., and Dopazo, J.: ‘A hierarchical unsupervised growing neural network for clustering gene expression patterns’, *Bioinformatics*, 2001, **17**, (2), pp. 126–136

47 Ressom, H., Wang, D., and Natarajan, P.: ‘Clustering gene expression data using adaptive double self-organizing map’, *Physiol. Genomics*, 2003, **14**, pp. 35–46

48 Ritchie, M.D., White, B.C., Parker, J.S., Hahn, L.W., and Moore, J.H.: ‘Optimization of neural network architecture using genetic programming improves detection and modeling of gene-gene interactions in studies of human diseases’, *BMC Bioinformatics*, 2003, **4**, (1), pp. 28–36

49 Dopazo, J., and Carazo, J.M.: 'Phylogenetic reconstruction using an unsupervised growing neural network that adopts the topology of a phylogenetic tree', *J. Mol. Evol.*, 1997, **44**, pp. 226–233

50 Parbhane, R.V., Tambe, S.S., and Kulkarni, B.D.: 'Analysis of DNA curvature using artificial neural networks', *Bioinformatics*, 1998, **14**, (2), pp. 131–138

51 Draghici, S., and Potter, R.B.: 'Predicting HIV drug resistance with neural networks', *Bioinformatics*, 2003, **19**, (1), pp. 98–107

52 Spicker, J.S., Wikman, F., Lu, M.L., Cordon-Cardo, C., Workman, C., ORntoft, T.F., Brunak, S., and Knudsen, S.: 'Neural network predicts sequence of TP53 gene based on DNA chip', *Bioinformatics*, 2002, **18**, (8), pp. 1133–1134

53 Alvager, T., Graham, G., Hutchison, D., and Westgard, J.: 'Neural network method to analyse data compression in DNA and RNA sequences', *J. Chem. Inf. Comput. Sci.*, 1997, **37**, (2), pp. 335–337

54 Sherriff, A., and Ott, J.: 'Applications of neural networks for gene finding', *Adv. Genet.*, 2001, **42**, pp. 287–297

55 Pal, S.K., Polkowski, L., and Skowron, A.: 'Rough-neuro computing: A way of computing with words' (Springer, Berlin, 2003)

56 Pal, S.K., and Mitra, S.: 'Neuro-fuzzy pattern recognition: methods in soft computing paradigm' (John Wiley, NY, 1999)

57 Pal, S.K., and Shiu, S.C.K.: 'Foundations of soft case based reasoning' (John Wiley, NY, 2004)