

18 Neural Networks in Bioinformatics

Ke Chen¹ · Lukasz A. Kurgan²

¹Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada

kchen1@ece.ualberta.ca

²Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada

lkurgan@ece.ualberta.ca

1	<i>Introduction</i>	566
2	<i>Biological Background</i>	567
3	<i>Neural Network Architectures in Protein Bioinformatics</i>	569
4	<i>Applications of Neural Networks in Protein Bioinformatics</i>	573
5	<i>Summary</i>	579

Abstract

Over the last two decades, neural networks (NNs) gradually became one of the indispensable tools in bioinformatics. This was fueled by the development and rapid growth of numerous biological databases that store data concerning DNA and RNA sequences, protein sequences and structures, and other macromolecular structures. The size and complexity of these data require the use of advanced computational tools. Computational analysis of these databases aims at exposing hidden information that provides insights which help with understanding the underlying biological principles. The most commonly explored capability of neural networks that is exploited in the context of bioinformatics is prediction. This is due to the existence of a large body of raw data and the availability of a limited amount of data that are annotated and can be used to derive the prediction model. In this chapter we discuss and summarize applications of neural networks in bioinformatics, with a particular focus on applications in protein bioinformatics. We summarize the most often used neural network architectures, and discuss several specific applications including prediction of protein secondary structure, solvent accessibility, and binding residues.

1 Introduction

The term “bioinformatics” was coined relatively recently, that is, it did not appear in the literature until 1991 (Boguski 1998). However, the first studies that concerned the field of bioinformatics appeared already in the 1960s when the first protein and nucleic acid sequence database was established. The National Institutes of Health (NIH) defines bioinformatics as “research, development, or application of computational tools and approaches for expanding the use of biological, medical, behavioral, or health data, including those to acquire, store, organize, archive, analyze, or visualize such data” (NIH Working Definition of Bioinformatics and Computational Biology 2000). We note that bioinformatics is usually constrained to molecular genetics and genomics. In a review by Luscombe et al. (2001), this term is defined as “conceptualizing biology in terms of macromolecules (in the sense of physical chemistry) and then applying ‘informatics’ techniques (derived from disciplines such as applied math, computer science, and statistics) to understand and organize the information associated with these molecules, on a large-scale.” The key observations concerning the above definition are that bioinformatics research is interdisciplinary, that is, it requires knowledge of physics, biochemistry, and informatics, and that it concerns large-scale analysis, that is, only scalable computational methods can be used. Since bioinformatics spans a wide variety of research areas, that is, sequence analysis, genome annotation, evolutionary biology, etc., we are not able to discuss all these research topics. Instead, we concentrate on the approaches concerning protein bioinformatics, that is, the scope of this chapter is limited to the application of bioinformatics in protein-related topics.

The last two decades observed an increased interest in the application of machine learning techniques, and particularly artificial neural networks (NNs), in protein bioinformatics. The most common application of the NNs is prediction; we assume that prediction concerns targets that are both discrete and real valued. The popularity of NNs stems from two key advantages that distinguish them from many other machine-learning methods. First, after the NN model is trained, the use of the model to perform prediction is very efficient, that is, computations are fast. This allows for a high throughput prediction of massive amounts of

data, which is an inherent feature of a significant majority of bioinformatics projects. Second, NN-based models provide high-quality results for many prediction tasks, for example, the leading methods in protein secondary structure prediction and protein solvent accessibility prediction are based on NNs. These successful applications raised the profile of NNs, which are currently being applied in dozens of other prediction tasks.

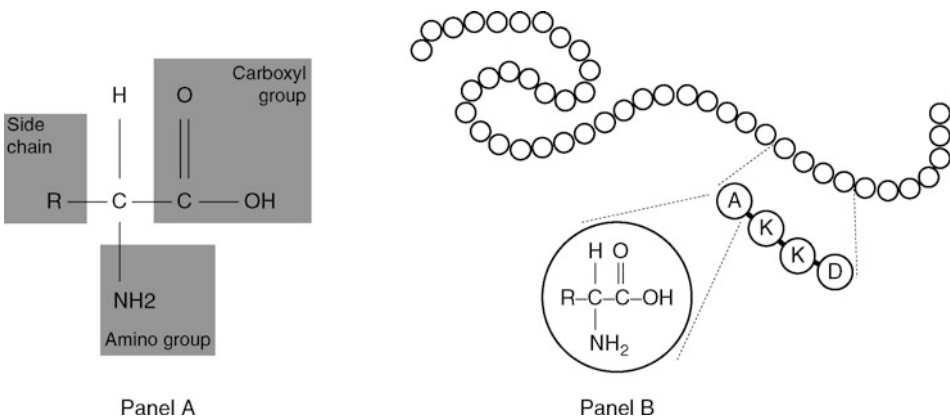
First, we introduce the relevant biological background. Next, we summarize the most popular NN architectures that are applied in protein bioinformatics and the key prediction methods that utilize NNs. Finally, we provide a more detailed analysis of NN-based solutions for the prediction of protein secondary structure, solvent accessibility, and binding residues.

2 Biological Background

Proteins are essential elements of virtually all living organisms. They participate in every process within cells. For instance, some proteins serve as enzymes that catalyze biochemical reactions which are vital to metabolism. Proteins are also important in cell signaling, immune responses, cell adhesion, and the cell cycle, to name just a few of their functions. They are large polymeric organic molecules which are composed of amino acids (also called residues). Amino acid (AA) is a small molecule that includes an amino ($-\text{NH}_2$) (except the proline amino acid) and a carboxyl ($-\text{COOH}$) group that are linked to a carbon atom. The AA formula, $\text{NH}_2\text{CHRCOOH}$, where N, H, C, and O are the nitrogen, hydrogen, carbon, and oxygen atoms, respectively, also incorporates R which denotes an organic substituent (so-called side chain), see [Fig. 1](#) (Panel A). There are a total of 20 AAs that make up all proteins. They all share the same NH_2CHCOOH group and have different R-group. The side chains determine physiochemical properties, such as charge, weight, and hydrophobicity, of

■ Fig. 1

Panel A shows the chemical structure of AAs; the side chain (R-group) differentiates the structure of different AAs. Panel B shows a protein chain (linear sequence) composed of AAs where each circle represents one AA.



- *Tertiary structure* defines the overall three-dimensional shape of a single protein molecule. It concerns the spatial arrangement of the secondary structures and is represented by the coordinates of all atoms in the protein. It is generally believed that the tertiary structure of a given protein is defined by its primary sequence and that each protein has a unique tertiary structure.
- *Quaternary structure* is the arrangement of multiple protein structures in a multi-subunit complex. The individual proteins are assembled into a larger molecule usually with a given geometrical shape, for example, protofilament or a spherical shape. For instance, a microtubule is the assembly of α -tubulin and β -tubulin proteins which takes the form of a hollow cylindrical filament.

While as of January 2009 the primary structure is known for over 6.4 million nonredundant proteins, the corresponding structure is known for only about 55,000 proteins. We emphasize that knowledge of the structure is of pivotal importance for learning and manipulating a protein's function, which for instance is exploited in modern drug design. The significant and widening gap between the set of known protein sequences and protein structures motivates the development of machine learning models that use the known structures to predict structures for the unsolved sequences.

3 Neural Network Architectures in Protein Bioinformatics

Although more than a dozen NN architectures have been developed and adopted, one of the first and simplest architectures, the feedforward neural network (FNN), is the most frequently applied in protein bioinformatics. Besides FNN, the recurrent neural network (RNN) and the radial basis function neural network (RBF) architectures also found several applications in the prediction of bioinformatics data.

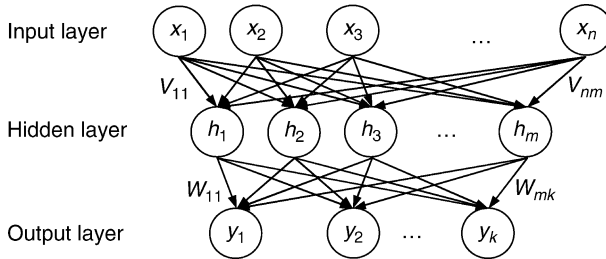
A common feature of all prediction applications in protein bioinformatics is the necessity to convert the input (biological) data into the data that can be processed by the NN. This usually involves encoding of the biological data into a fixed-size feature vector. For instance, the primary protein structure is represented as a variable length string of characters with an alphabet of 20 letters (AAs), see [▶ Fig. 2](#) (Panel A). This sequence is converted into a vector of numerical features that constitutes the input to the NN. For instance, the vector could include 20 counts of the occurrence of the 20 amino acids in the sequence. The following discussion assumes that the input data are already encoded into the feature vector.

3.1 Feedforward Neural Networks

The FNN architecture usually consists of three layers, an input layer, a hidden layer, and an output layer. The input layer accepts the input feature vector and the output layer generates the predictions. The hidden layer is responsible for capturing the prediction model. Each layer consists of a number of nodes and each node in a given layer connects with every other node in the following layer, see [▶ Fig. 3](#). The connections are associated with weights v_{ij} and w_{ij} between the i th node in one layer and the j th node in the next layer. The nodes process the input values, which are computed as the weighted sum of values passed from the previous layer, using activation functions. The two most frequently used activation functions are:

■ Fig. 3

Architecture of FNN. The input layer contains n nodes (which equals the number of features in the input feature vector), the hidden layer contains m nodes and the output layer contains k nodes. The weight between the i th node of the input layer and the j th node of the hidden layer is denoted by v_{ij} . The weight between the i th node of the hidden layer and the j th node of the output layer is denoted by w_{ij} .



$$\begin{aligned}\phi(v_i) &= \tanh(v_i) \\ \phi(v_i) &= (1 + e^{-v_i})^{-1}\end{aligned}$$

where v_i is the weighted sum of the inputs. The values of the former hyperbolic tangent function range between -1 and 1 , while the values of the latter logistic function range between 0 and 1 . Some applications also utilize radial basis activation functions.

Learning using the FNN-based prediction model is performed by adjusting the connection weight values to minimize the prediction error on training data. For a given input feature vector $\{x_i\}$, the observations (the prediction outcomes) are denoted as $\{y_i\}$. The goal of the FNN is to find a function $f: X \rightarrow Y$, which describes the relation between inputs X and observations Y . The merit of function f is measured with a cost function $C = E[(f(x_i) - y_i)^2]$. For a training dataset with n samples, the cost function is

$$C = \sum_{i=1}^n \frac{[f(x_i) - y_i]^2}{n}$$

Based on the amount of error associated with the outputs of the network in comparison with the expected result (cost function), the adjustment of the connection weights is carried out using a backpropagation algorithm. The n input feature vectors are fed multiple times (each presentation of the entire training dataset is called an epoch) until the weight values do not change or a desired value of the cost function is obtained.

FNN is the most widely applied among the NN architectures in protein bioinformatics. The applications include:

- Prediction of the secondary structure of protein (Jones 1999; Rost et al. 1994; Dor and Zhou 2007a; Hung and Samudrala 2003; Petersen et al. 2000; Qian and Sejnowski 1988). The aim of these methods is to predict the secondary structure state (helix, strand, or coil) for every AA in the input protein sequence.
- Prediction of relatively solvent accessibility of protein residues (Rost and Sander 1994; Garg et al. 2005; Adamczak et al. 2005; Ahmad et al. 2003; Dor and Zhou 2007b; Pollastri

et al. 2002a). The solvent accessibility is defined as a fraction of a surface area of a given AA that is accessible to the solvent. The AAs with high solvent accessibility are usually on the protein surface.

- Prediction of binding residues (Jeong et al. 2004; Ahmad and Sarai 2005; Zhou and Shan 2001; Ofra and Rost 2007). The binding residues are those AAs in a given protein that are involved in interactions with another molecule. The interactions could concern other proteins, DNA, RNA, ions, etc., and they usually implement protein functions.
- Prediction of transmembrane regions (Gromiha et al. 2005; Natt et al. 2004; Jacoboni et al. 2001). Some proteins are embedded into cell membranes and they serve as pumps, channels, receptors, and energy transducers for the cell. The goal of this prediction method is to find which AAs in the input protein sequence are embedded into the membrane.
- Prediction of subcellular location of proteins (Zou et al. 2007; Cai et al. 2002; Reinhardt and Hubbard 1998; Emanuelsson et al. 2000). These methods predict the location of the proteins inside a cell. The locations include cytoplasm, cytoskeleton, endoplasmic reticulum, Golgi apparatus, mitochondrion, nucleus, etc.

3.2 Recurrent Neural Networks

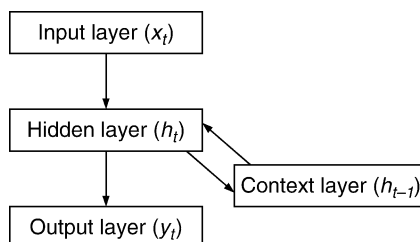
A recurrent neural network (RNN) is a modification to the FNN architecture. In this case, a “context” layer is added, and this layer retains information across observations. In each iteration, a new feature vector is fed into the input layer. The previous contents of the hidden layer are copied to the context layer and then fed back into the hidden layer in the next iteration, see [Fig. 4](#).

When an input feature vector is fed into the input layer, the RNN processes are as follows:

1. Copy the input vector values into the input nodes.
2. Compute hidden node activations using net input from input nodes and from the nodes in the context layer.
3. Compute output node activations.
4. Compute the new weight values using the backpropagation algorithm.
5. Copy new hidden node weights to the context layer.

■ Fig. 4

Architecture of RNN. Like FNN, RNN also contains an input layer, a hidden layer, and an output layer. An additional context layer is connected to the hidden layer.



Since the trainable weights, that is, weights between the input and hidden layers and between the hidden and output layers, are feedforward only, the standard backpropagation algorithm is applied to learn the weight values. The weights between the context and the hidden layers play a special role in the computation of the cost function. The error values they receive come from the hidden nodes and so they depend on the error at the hidden nodes at the t th iteration. During the training of the RNN model we consider a gradient of a cost (error) function which is determined by the activations at both the present and the previous iterations.

The RNN architecture was successfully applied in the prediction of:

- Beta-turns (Kirschner and Frishman 2008). Beta turns are the most frequent subtypes of coils, which are one of the secondary protein structures.
- Secondary structure of proteins (Chen and Chaudhari 2007).
- Continuous B-cell epitopes (Saha and Raghava 2006). B-cell epitopes are the antigenic regions of proteins recognized by the binding sites of immunoglobulin molecules. They play an important role in the development of synthetic vaccines and in disease diagnosis. The goal of this prediction method is to find AAs that correspond to the epitopes.
- Number of residue contacts (Pollastri et al. 2002b), which is defined as the number of contacts a given AA makes in the three-dimensional protein molecule. The knowledge of the contacts helps in learning the tertiary protein structure.

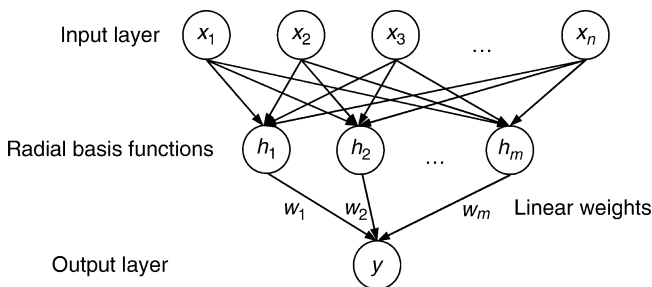
3.3 Radial Basis Function Neural Networks

Radial basis function (RBF) neural networks also incorporate three layers: an input layer, a hidden layer with a nonlinear RBF activation function, and a linear output layer, see [Fig. 5](#). During the process of training the RBF model:

1. The input vectors are mapped onto each RBF in the hidden layer. The RBF is usually implemented as a Gaussian function. The Gaussian functions are parameterized, that is, values of the center and spread are established, using the training dataset. The commonly

■ Fig. 5

Architecture of RBF neural network. The network is fully connected between the input and the hidden layers (each node in the input layer is connected with all nodes in the hidden layer), and all the weights are usually assumed to be equal to 1. The nodes in the hidden layer are fully connected with a single node in the output layer, and the weight values are optimized to minimize the cost function.



used methods include K-means clustering or, alternatively, a random subset of the training vectors can be used as the centers.

2. In regression problems (the prediction outcomes are real values), the output layer is a linear combination of values produced by the hidden layer, which corresponds to the mean predicted output. In prediction problems (the prediction outcomes are discrete), the output layer is usually implemented using a sigmoid function of a linear combination of hidden layer values, representing a posterior probability.

RBF networks are faster to train when compared with FNN and RNN. They also have an advantage of not suffering from local minima in the same way as FNN, that is, the FNN may not be able to find globally best solution, but it may get stuck in a local minimum of the cost function. This is because the only parameters that are adjusted in the learning process of the RBF network are associated with the linear mapping from the hidden layer to the output layer. The linearity ensures that the error surface is quadratic and therefore it has a single, usually relatively easy to find, minimum. At the same time, the quality of the prediction is usually higher when using a properly designed and trained FNN.

RBF networks were utilized in several applications that include:

- Prediction of inter-residue contact maps (Zhang and Huang 2004). The contact maps include binary entries that define whether a given AA is or is not in contact with any other AA in the tertiary structure. The knowledge of contacts helps in the reconstruction of the tertiary protein structure.
- Prediction of protease cleavage sites (Yang and Thomson 2005). Protease cleavage is performed by enzymes, which are proteins that catalyze biological reactions. Knowledge of how a given protease cleaves the proteins is important for designing effective inhibitors to treat some diseases. This prediction method aims at finding AAs in the protein sequence that are involved in this interaction.
- Prediction of targets for protein-targeting compounds (Niwa 2004). This method aims at the prediction of biological targets (proteins) that interact with given chemical compounds. This has applications in drug design where large libraries of chemical compounds are screened to find compounds that interact with a given protein and which, as a result, modify (say, inhibit) the protein's function.

One of the important parameters in the design of any of the three above mentioned architectures, that is, FNN, RNN, and RBFNN, is the number of nodes. The number of input nodes usually equals the number of input features. Most commonly, there is only one output node that corresponds to the predicted outcome, although in some cases NNs are used to generate multiple outcomes simultaneously, that is, prediction of the protein secondary structure requires three outcomes. The number of nodes in the hidden layer is chosen by the designer of the network. This number depends on the application and the desired quality of the prediction.

4 Applications of Neural Networks in Protein Bioinformatics

NNs are used in a variety of protein bioinformatics applications. They can be categorized into:

- Prediction of protein structure including secondary structure and secondary structure content, contact maps, structural contacts, boundaries of structural domains, specific types of local structures like beta-turns, etc.

- Prediction of binding sites and ligands, which includes prediction of binding residues and prediction of various properties of the binding ligands.
- Prediction of protein properties such as physicochemical proteins, localization in the host organism, etc.


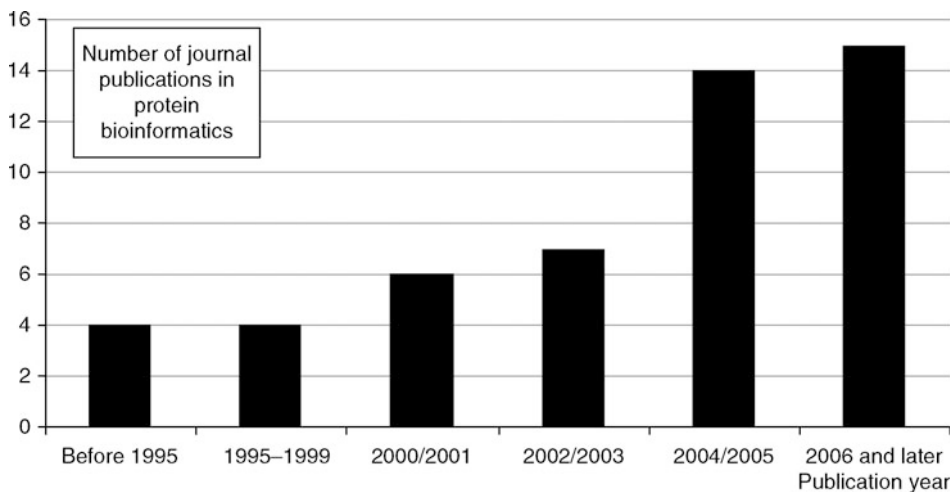
Specific applications include prediction of a number of residue contacts (Pollastri et al. 2002), protein contact maps (Zhang and Huang 2004), helix-helix (Fuchs et al. 2009) and disulfide contacts (Martelli et al. 2004), beta and gamma turns (Kirschner and Frishman 2008; Kaur and Raghava 2003, 2004), secondary structure (Jones 1999; Rost et al. 1994; Dor and Zhou 2007a; Hung and Samudrala 2003; Petersen et al. 2000; Qian and Sejnowski 1988; Chen and Chaudhari 2007), domain boundaries (Ye et al. 2008), transmembrane regions (Gromiha et al. 2005; Natt et al. 2004; Jacoboni et al. 2001), binding sites and functional sites (Jeong et al. 2004; Ahmad and Sarai 2005; Zhou and Shan 2001; Ofraan and Rost 2007; Yang and Thomson 2005; Lin et al. 2005; Lundegaard et al. 2008; Blom et al. 1996; Ingrell et al. 2007), residue solvent accessibility (Rost and Sander 1994; Garg et al. 2005; Adamczak et al. 2005; Ahmad et al. 2003; Dor and Zhou 2007b; Pollastri et al. 2002), subcellular location (Zou et al. 2007; Cai et al. 2002; Reinhardt and Hubbard 1998; Emanuelsson et al. 2000), secondary structure content (Muskal and Kim 1992; Cai et al. 2003; Ruan et al. 2005), backbone torsion angles (Xue et al. 2008; Kuang et al. 2004), protein structural class (Chandonia and Karplus 1995; Cai and Zhou 2000), signal peptides (Plewczynski et al. 2008; Sidhu and Yang 2006), continuous B-cell epitopes (Saha and Raghava 2006), binding affinities, toxicity, and pharmacokinetic parameters of organic compounds (Vedani and Dobler 2000), biological targets of chemical compounds (Niwa 2004), and prediction of spectral properties of green fluorescent proteins (Nantasenamat et al. 2007). We observe a growing interest in applying NNs in this domain, see  Fig. 6. The NN-based applications in protein bioinformatics were published in a number of

 Fig. 6


Number of journal publications (y-axis) concerning the applications of NNs in protein bioinformatics in the last two decades. The included publications do not constitute an exhaustive list of corresponding studies, but rather they provide a set of the most significant and representative developments.




high-impact scientific journals such as (in alphabetical order) Bioinformatics; BMC Bioinformatics; Gene; IEEE/ACM Transactions on Computational Biology and Bioinformatics; Journal of Computational Chemistry; Journal of Computer-Aided Molecular Design; Journal of Medicinal Chemistry; Journal of Molecular Biology; Nucleic Acids Research; PLoS Computational Biology; Protein Science; Proteins; and Proteomics. The number, scope, and quality of the above venues strongly indicate the important role of this research.

The prediction of the secondary structure, residue solvent accessibility, and binding sites attracted the most attention in the context of the NN-based solutions. Therefore, the following sections concentrate on these three topics.

4.1 Prediction of Protein Secondary Structure with Neural Networks

Protein secondary structure is defined as a regular and repetitive spatially local structural pattern in protein structures. Several methods are used to define the protein secondary structure from a protein's three-dimensional structure. The most commonly used method is the Dictionary of Protein Secondary Structure (DSSP) (Kabsch and Sander 1983), which assigns eight types of secondary structures based on hydrogen-bonding patterns. These types include 3/10 helix, alpha helix, pi helix, extended strand in parallel and/or antiparallel β -sheet conformation, isolated β -bridge, hydrogen bonded turn, bend, and coil. The eight-state secondary structure is often aggregated into a three-state secondary structure. The first three types are combined into the helix state, the following two types into the strand state, and the last three types into the coil state. Most of the existing computational methods predict the three-state secondary structure instead of the eight-state structure. The main goal of these methods is to obtain the secondary structure using only AA sequences of the protein as the input.  *Figure 2* shows the AA sequence, the corresponding secondary structure for each AA, the spatial arrangement of the secondary structure, and the overall three-dimensional structure of the human prion protein. This protein is associated with several prion diseases such as fatal familial insomnia and Creutzfeldt–Jakob disease.

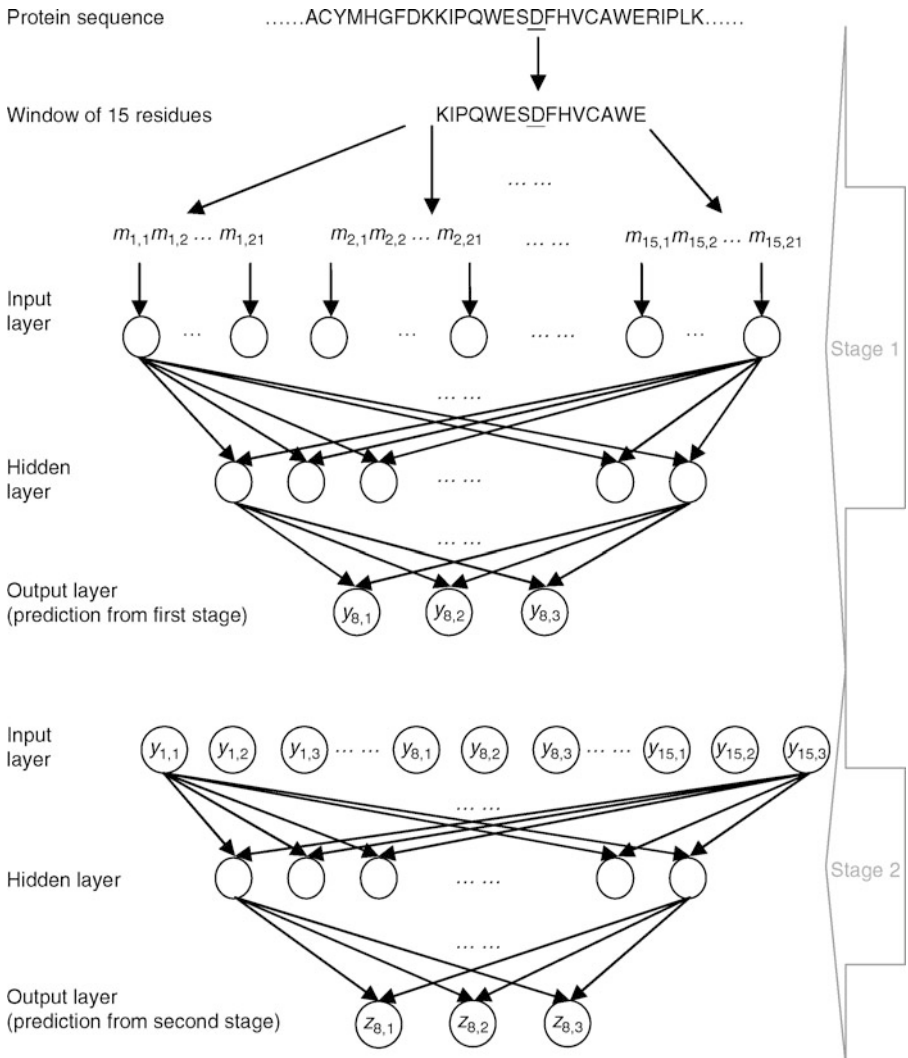
The first study concerning the prediction of protein secondary structure using an NN appeared in 1988 (Qian and Sejnowski 1988). This model is a typical three-layer FNN in which the input layer contains $13 \times 21 = 273$ nodes representing a stretch of 13 continuous AAs in the sequence, and the output layer contains three nodes representing the three secondary structure states. Each AA in the sequence is encoded using 21 binary features indicating the type of the AA at a given position in the sequence. This early method was trained using a very small dataset of 106 protein sequences, which limited its quality.

One of the most successful and commonly used models for the prediction of protein secondary structure, named PSIPRED, was proposed by Jones in 1999 (Jones 1999). It is a two-stage NN that takes a position-specific scoring matrix (PSSM), which is generated from the protein sequence using the PSI-BLAST (Position Specific Iterated Basic Local Alignment Search Tool) algorithm (Altschul et al. 1997) as the input. The architecture of PSIPRED is summarized in  *Fig. 7*.


In the first stage, the input protein sequence is represented by the PSSM using a window of size 15 which is centered over the predicted AA. PSSM includes 20 dimensions for each AA, which correspond to substitution scores for each of the 20 AAs. The scores quantify which AAs are likely to be present/absent at a given position in the sequence in a set of known sequences that are similar to the sequence being predicted. This is based on the assumption that

■ Fig. 7

Architecture of PSIPRED algorithm. The algorithm is two-stage and includes two 3-layer FNNs, where the output of the first stage network feeds into the input to the second stage network. In the first stage, a window of 15 positions over the PSSM profile generated by the PSI-BLAST program from the input protein sequence is used. Each position in the input is represented by a vector of 21 values (the i th AA in the window is represented as $m_{i,1}m_{i,2} \dots m_{i,21}$). The 21×15 values are fed into the input layer. The output layer in the first stage NN contains three nodes that represent the probabilities of forming helix, strand, and coil structures (the predicted probabilities for the central AA in the window are represented as $y_{8,1}$, $y_{8,2}$, and $y_{8,3}$). These probabilities, using a window of 15 positions, are fed into the second-stage NN. The output from the second-stage NN is the final prediction that represents the probabilities of three types of the secondary structure: $z_{8,1}$, $z_{8,2}$, and $z_{8,3}$.



similarity in the sequence often implies similarity in the structure. The positive scores indicate that a given AA substitution occurs more frequently than expected by chance, while negative scores indicate that the substitution occurs less frequently than expected. The 20 scores from the PSSM together with a single feature that indicates the terminus of the sequence are fed into the input layer of the first-stage NN. As a result, the input layer contains $15 \times 21 = 315$ nodes. The hidden layer contains 75 nodes and the output layer contains three nodes which indicate the probabilities of the three secondary structure states.

In the second stage, the predicted probabilities of the secondary structures from the first stage for a window of 15 AAs centered over the position being predicted are fed into the input layer. The second layer exploits the fact that secondary structures form segments in the protein sequence, see  Fig. 2a, and thus information about the structure of the AAs in the adjacent positions in the sequence is helpful to determine the structure of a given AA. The input layer contains $4 \times 15 = 60$ nodes (the value indicating the terminus of the sequence is also included), the hidden layer contains 60 nodes, and the output layer contains three nodes. The PSIPRED method can be accessed, as a web server, at <http://bioinf.cs.ucl.ac.uk/psipred/>. Interested users can also download a stand-alone version of this popular prediction method.

One of the recently proposed NN-based methods performs the prediction of the secondary structure using a cascaded bidirectional recurrent neural network (BRNN) (Chen and Chaudhari 2007). Similar to the PSIPRED design, the first BRNN (sequence-to-structure BRNN) predicts the secondary structure based on the input AA sequences. The second BRNN (structure-to-structure BRNN) refines the raw predictions from the first BRNN. The learning algorithm used to develop this method is based on the backpropagation.

The last two decades observed the development of several methods based on NN for the prediction of protein secondary structure. The performance of the methods mainly depends on the representation of the protein sequence and the size of the training dataset. Since the beta-sheets (strands adjacent in the tertiary structure) are established between AAs that are far away in the sequence, the window-based methods (including all present methods for the prediction of protein secondary structure) are inherently incapable of grasping the long-range interactions, which results in a relatively poor result for strands.

4.2 Prediction of Binding Sites with Neural Networks

A protein performs its function through interactions with other molecules, called ligands, which include another protein, DNA, RNA, small organic compounds, or metal ions. Knowledge of the binding sites, which are defined as the AAs that directly interact with the other molecules, is crucial to understand the protein's function. More specifically, an AA is a part of the binding site if the distance from at least one atom of this AA to any atom of the ligand is less than a cutoff threshold D . The values of D vary in different studies and they usually range between 3.5 and 6 Å (Zhou and Shan 2001; Ofra and Rost 2007; Ahmad et al. 2004; Kuznetsov et al. 2006).

In one of the recent works by Jeong and colleagues, the FNN architecture is used for the prediction of RNA-binding sites (Jeong et al. 2004). Each AA in the input protein sequence is encoded by a vector of 24 values, of which 20 values indicate the AA type (using binary encoding), one value represents the terminus of the sequence, and the remaining three values correspond to the probabilities of three types of the secondary structure predicted using the PHD program. Using a window of size of 41 residues, the corresponding design includes

$24 \times 41 = 984$ input nodes. The hidden layer includes 30 nodes and the output layer consists of a single node that provides the prediction.

In another recent study by Ahmad and Sarai, a three-layer FNN is utilized for the prediction of DNA-binding sites (Ahmad and Sarai 2005). The architecture of this model is relatively simple, that is, 100 nodes in the input layer, 2 nodes in the hidden layer, and 1 node in the output layer. The input layer receives values from the PSSM computed over the input protein sequence with the window size of 5, which results in 100 features per AA.

Prediction of protein–protein interaction sites uses designs that are similar to the designs utilized in the prediction of DNA/RNA-binding sites. Zhou and Shan proposed a three-layer FNN to predict the protein–protein interaction sites (Zhou and Shan 2001). In their design, the PSSM and solvent-accessible area generated by the DSSP program (Kabsch and Sander 1983) for the predicted AAs and the 19 spatially nearest neighboring surface AAs make up the input. As a result, the input layer contains $21 \times 20 = 420$ nodes. The hidden layer includes 75 nodes. This method predicts the protein–protein interaction sites from the protein’s three-dimensional structure, since this information is necessary to compute the relative solvent accessibility values and to find the 19 spatially nearest AAs. In a recent study by Ofran and Rost, a classical FNN model is used for the prediction of protein–protein interaction sites from the protein sequence (Ofra and Rost 2007). In this case, AAs in the input protein sequence are represented using PSMM, predicted values of solvent accessibility, predicted secondary structure state, and a conservation score. The window size is set to include eight AAs surrounding the position that is being predicted, and the above-mentioned information concerning these nine amino acids is fed into the input layer.

NNs were also applied for prediction of metal-binding sites (Lin et al. 2005), binding sites for a specific protein family, that is, the binding sites of MHC I (Lundegaard et al. 2008), and prediction of functional sites, that is, the cleavage sites (Blom et al. 1996) and phosphorylation sites (Ingrell et al. 2007).


4.3 Prediction of Relative Solvent Accessibility with Neural Networks

Relative solvent accessibility (RSA) reflects the percentage of the surface area of a given AA in the protein sequence that is accessible to the solvent. RSA value, which is normalized to the $[0, 1]$ interval, is defined as the ratio between the solvent accessible surface area (ASA) of an AA within a three-dimensional structure and the ASA of its extended tripeptide (Ala-X-Ala) conformation:

$$\text{RSA} = \frac{\text{RSA in 3D structure}}{\text{RSA in extended tripeptide conformation}}$$

The first study that concerned prediction of RSA from the protein sequence was published in 1994 by Rost and Sander (Rost and Sander 1994). In this work, the AA is encoded by the percentage of the occurrence of each AA type at this position in the sequence in multiple sequence alignment, which is similar to the values provided in the PSSM matrix. The input to the two-layer FNN is based on a window of size 9 which is centered on the AA that is being predicted and is used, which results in 9×20 features, together with the AA composition of the entire protein sequence, length of the sequence (using four values), and distance of the window from two termini of the sequence (using four values for each terminus). As a result, the network has a total of $180 + 20 + 4 + 8 = 212$ nodes in the input

layer. The output layer contains one node representing the predicted RSA value and no hidden layer is used in this model.

The past decade observed the development of several NN-based methods for the prediction of RSA values (Garg et al. 2005; Adamczak et al. 2005; Ahmad et al. 2003; Dor and Zhou 2007b; Pollastri et al. 2002). These methods share similar architectures and therefore we discuss one representative model proposed by Garg et al. (2005). This method is a two-stage design in which both stages are implemented using FNNs. Two sources of information are used to generate inputs for the NNs from the protein sequence, the PSSM profile, and the secondary structure predicted with the PSIPRED algorithm. The input features are extracted using a window of size 11 centered on the AA that is being predicted. The values from PSSM in the window are fed into the first NN. This results in the input layer with $11 \times 21 = 231$ nodes. The hidden layer contains ten nodes and the output layer has one node. In the second stage NN, the predicted RSA values of the AAs in the window and the predicted probabilities of the three secondary structure types predicted by PSIPRED in the same window are fed into the input layer. This results in $11 \times 4 = 44$ nodes in the input layer. The hidden layer includes ten nodes and the single node in the output layer corresponds to the final prediction. The architecture of this method is shown in  Fig. 8. The second layer exploits the observation that information about the secondary structure and solvent accessibility of the AAs in the adjacent positions in the sequence is useful in determining the solvent accessibility of a given AA. We observe that a similar design is used to implement the PSIPRED method.

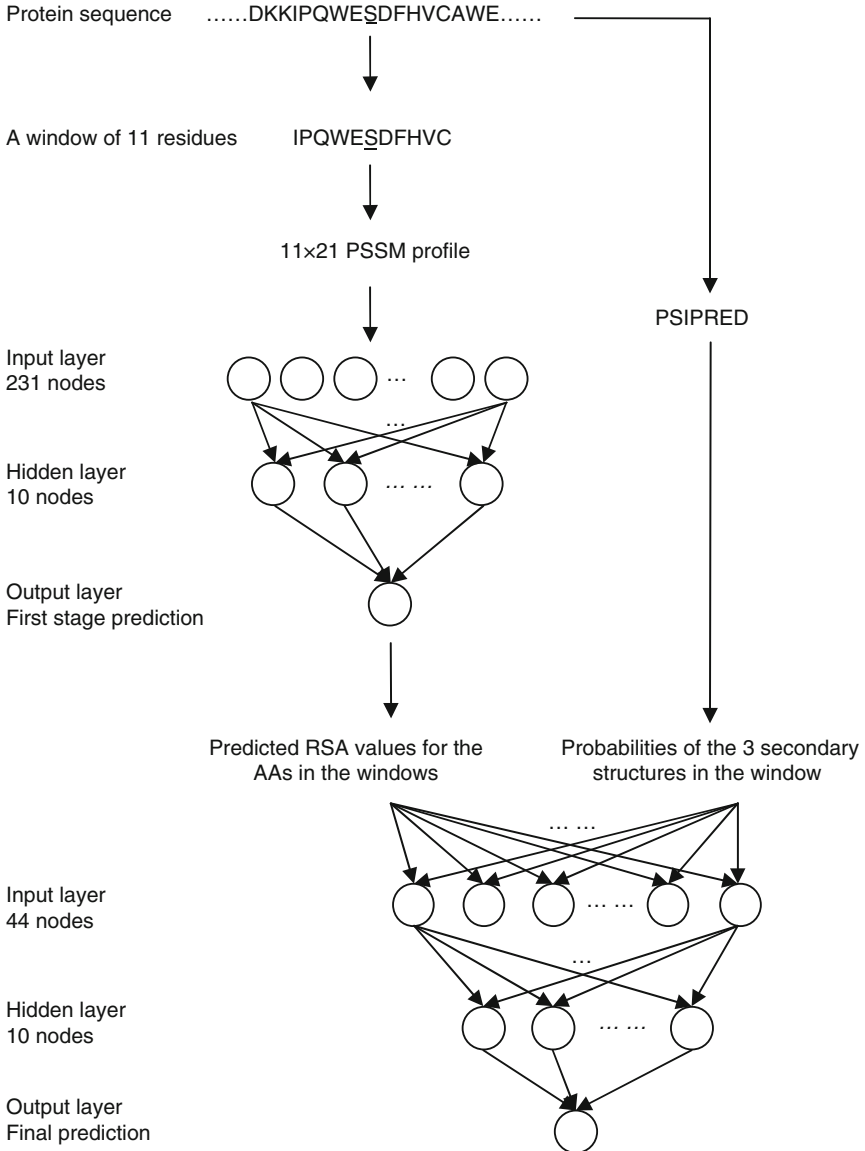
5 Summary

We summarized the applications of neural networks (NN) in bioinformatics, with a particular focus on protein bioinformatics. We show that numerous applications that aim at predictions of a variety of protein-related information, such as structure, binding sites, and localization, are designed and implemented using NNs. The most popular architecture used in these methods is a simple three-layer feedforward NN, although other architectures such as RBF and recurrent NNs are also applied. Some of the protein bioinformatics applications use multilayered designs in which two (or more) NNs are used in tandem. We show that the popularity of the NN-based designs has been growing over the last decade. Three applications that enjoy the most widespread use are discussed in greater detail. They include protein secondary structure prediction, prediction of binding sites, and prediction of relative solvent accessibility. We contrast and analyze the architectures of the corresponding NN models. We conclude that the extent and quality of the applications that are based on NNs indicate that this methodology provides sound and valuable results for the bioinformatics community.

We acknowledge several other useful resources that discuss the applications of a broader range of machine learning techniques in bioinformatics. Although none of these contributions is solely devoted to NNs, some of them discuss NNs together with other similar techniques. A survey by Narayanan and colleagues discusses applications of classification methods (nearest neighbor and decision trees), NNs, and genetic algorithms in bioinformatics (Narayanan et al. 2002). Another survey contribution by Kapetanovic and coworkers concerns clustering and classification algorithms, including NNs and support vector machines (Kapetanovic et al. 2004). The most recent review by Fogel discusses a host of computational intelligence techniques, including NNs, fuzzy systems, and evolutionary computation, in the context of

■ Fig. 8

Architecture of the model proposed in Garg et al. (2005) for the prediction of relative solvent accessibility. The method includes two stages implemented using FNNs. In the first stage, a window of 11 AAs is used, and each AA is represented by a vector of 21 values. The vector is taken from the PSSM profile generated by the PSI-BLAST algorithm. The output layer of the first stage generates one value that represents the predicted RSA value, which is further refined using the second stage. The predicted RSA values and the secondary structure probabilities predicted using PSIPRED of the AAs in the window are fed into the second-stage FNN. The output from the second-stage NN constitutes the final predicted RSA value.



bioinformatics (Fogel 2008). Several other surveys that do not treat NNs but which focus on the use of other related techniques in bioinformatics were published in recent years. They include a paper by Byvatov and Schneider (2003) that concerns applications of support vector machines; a contribution by Saeys et al. (2007) that discusses feature selection methods; a survey concerning Bayesian networks by Wilkinson (2007); a review of supervised classification, clustering, and probabilistic graphical models by Larranaga et al. (2006); and a recent contribution by Miller et al. (2008) that focuses on clustering.

References

- Adamczak R, Porollo A, Meller J (2005) Combining prediction of secondary structure and solvent accessibility in proteins. *Proteins* 59:467–475
- Ahmad S, Gromiha MM, Sarai A (2003) Real value prediction of solvent accessibility from amino acid sequence. *Proteins* 50:629–635
- Ahmad S, Gromiha MM, Sarai A (2004) Analysis and prediction of DNA-binding proteins and their binding residues based on composition, sequence and structural information. *Bioinformatics* 20:477–486
- Ahmad S, Sarai A (2005) PSSM-based prediction of DNA binding sites in proteins. *BMC Bioinformatics* 6:33
- Altschul SF, Madden TL, Schaffer AA, Zhang JH, Zhang Z, Miller W, Lipman DJ (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 17:3389–3402
- Blom N, Hansen J, Blaas D, Brunak S (1996) Cleavage site analysis in picornaviral polyproteins: discovering cellular targets by neural networks. *Protein Sci* 5:2203–2216
- Boguski MS (1998) Bioinformatics – a new era. *Trends Guide Bioinformatics (Suppl S)*:1–3
- Byvatov E, Schneider G (2003) Support vector machine applications in bioinformatics. *Appl Bioinformatics* 2(2):67–77
- Cai YD, Zhou GP (2000) Prediction of protein structural classes by neural network. *Biochimie* 82:783–785
- Cai YD, Liu XJ, Chou KC (2002) Artificial neural network model for predicting protein subcellular location. *Comput Chem* 26:179–182
- Cai YD, Liu XJ, Chou KC (2003) Prediction of protein secondary structure content by artificial neural network. *J Comput Chem* 24:727–731
- Chandonia JM, Karplus M (1995) Neural networks for secondary structure and structural class predictions. *Protein Sci* 4:275–285
- Chen J, Chaudhari N (2007) Cascaded bidirectional recurrent neural networks for protein secondary structure prediction. *IEEE/ACM Trans Comput Biol Bioinform* 4:572–582
- Dor O, Zhou Y (2007a) Achieving 80% ten-fold cross-validated accuracy for secondary structure prediction by large-scale training. *Proteins* 66:838–845
- Dor O, Zhou Y (2007b) Real-SPINE: an integrated system of neural networks for real-value prediction of protein structural properties. *Proteins* 68:76–81
- Emanuelsson O, Nielsen H, Brunak S, von Heijne G (2000) Predicting subcellular localization of proteins based on their N-terminal amino acid sequence. *J Mol Biol* 300:1005–1016
- Fogel GB (2008) Computational intelligence approaches for pattern discovery in biological systems. *Brief Bioinform* 9(4):307–316
- Fuchs A, Kirschner A, Frishman D (2009) Prediction of helix-helix contacts and interacting helices in polytopic membrane proteins using neural networks. *Proteins* 74:857–871
- Garg A, Kaur H, Raghava GP (2005) Real value prediction of solvent accessibility in proteins using multiple sequence alignment and secondary structure. *Proteins* 61:318–324
- Gromiha MM, Ahmad S, Suwa M (2005) TMBETA-NET: discrimination and prediction of membrane spanning beta-strands in outer membrane proteins. *Nucleic Acids Res* 33:W164–167
- Hung LH, Samudrala R (2003) PROTFINFO: secondary and tertiary protein structure prediction. *Nucleic Acids Res* 31:3296–3299
- Ingrell CR, Miller ML, Jensen ON, Blom N (2007) Net-PhosYeast: prediction of protein phosphorylation sites in yeast. *Bioinformatics* 23:895–897
- Jacoboni I, Martelli PL, Fariselli P, De Pinto V, Casadio R (2001) Prediction of the transmembrane regions of beta-barrel membrane proteins with a neural network-based predictor. *Protein Sci* 10:779–787
- Jeong E, Chung IF, Miyano S (2004) A neural network method for identification of RNA-interacting residues in protein. *Genome Inform* 15:105–116
- Jones DT (1999) Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol* 292:195–202
- Kabsch W, Sander C (1983) Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers* 22:2577–2637

- Kapetanovic IM, Rosenfeld S, Izmirlian G (2004) Overview of commonly used bioinformatics methods and their applications. *Ann NY Acad Sci* 1020:10–21
- Kaur H, Raghava GP (2003) A neural-network based method for prediction of gamma-turns in proteins from multiple sequence alignment. *Protein Sci* 12:923–929
- Kaur H, Raghava GP (2004) A neural network method for prediction of beta-turn types in proteins using evolutionary information. *Bioinformatics* 20:2751–2758
- Kirschner A, Frishman D (2008) Prediction of beta-turns and beta-turn types by a novel bidirectional Elman-type recurrent neural network with multiple output layers (MOLEBRNN). *Gene* 422:22–29
- Kuang R, Leslie CS, Yang AS (2004) Protein backbone angle prediction with machine learning approaches. *Bioinformatics* 20:1612–1621
- Kuznetsov IB, Gou Z, Li R, Hwang S (2006) Using evolutionary and structural information to predict DNA-binding sites on DNA-binding proteins. *Proteins* 64:19–27
- Larranaga P, Calvo B, Santana R, Bielza C, Galdiano J, Inza I, Lozano JA, Armananzas R, Santafe G, Perez A, Robles V (2006) Machine learning in bioinformatics. *Brief Bioinformatics* 7(1):86–112
- Lin CT, Lin KL, Yang CH, Chung IF, Huang CD, Yang YS (2005) Protein metal binding residue prediction based on neural networks. *Int J Neural Syst* 15:71–84
- Lundegaard C, Lamberth K, Harndahl M, Buus S, Lund O, Nielsen M (2008) NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8–11. *Nucleic Acids Res* 36:W509–512
- Luscombe NM, Greenbaum D, Gerstein M (2001) What is bioinformatics? A proposed definition and overview of the field. *Methods Inf Med* 40:346–358
- Martelli PL, Fariselli P, Casadio R (2004) Prediction of disulfide-bonded cysteines in proteomes with a hidden neural network. *Proteomics* 4:1665–1671
- Miller DJ, Wang Y, Kesidis G (2008) Emergent unsupervised clustering paradigms with potential application to bioinformatics. *Front Biosci* 13:677–690
- Muskal SM, Kim SH (1992) Predicting protein secondary structure content. A tandem neural network approach. *J Mol Biol* 225:713–727
- Nantasenamat C, Isarankura-Na-Ayudhya C, Tansila N, Naenna T, Prachayasittikul V (2007) Prediction of GFP spectral properties using artificial neural network. *J Comput Chem* 28:1275–1289
- Narayanan A, Keedwell EC, Olsson B (2002) Artificial intelligence techniques for bioinformatics. *Appl Bioinformatics* 1(4):191–222
- Natt NK, Kaur H, Raghava GP (2004) Prediction of transmembrane regions of beta-barrel proteins using ANN- and SVM-based methods. *Proteins* 56:11–18
- NIH Working Definition of Bioinformatics and Computational Biology (2000) BISTIC Definition Committee, <http://www.bisti.nih.gov/>
- Niwa T (2004) Prediction of biological targets using probabilistic neural networks and atom-type descriptors. *J Med Chem* 47:2645–2650
- Ofran Y, Rost B (2007) Protein-protein interaction hotspots carved into sequences. *PLoS Comput Biol* 3:e119
- Petersen TN, Lundegaard C, Nielsen M, Bohr H, Bohr J, Brunak S, Gippert GP, Lund O (2000) Prediction of protein secondary structure at 80% accuracy. *Proteins* 41:17–20
- Plewczynski D, Slabinski L, Ginalski K, Rychlewski L (2008) Prediction of signal peptides in protein sequences by neural networks. *Acta Biochim Pol* 55:261–267
- Pollastri G, Baldi P, Fariselli P, Casadio R (2002a) Prediction of coordination number and relative solvent accessibility in proteins. *Proteins* 47:142–153
- Pollastri G, Baldi P, Fariselli P, Casadio R (2002b) Prediction of coordination number and relative solvent accessibility in proteins. *Proteins* 47:142–153
- Qian N, Sejnowski TJ (1988) Predicting the secondary structure of globular proteins using neural network models. *J Mol Biol* 202:865–884
- Reinhardt A, Hubbard T (1998) Using neural networks for prediction of the subcellular location of proteins. *Nucleic Acids Res* 26:2230–2236
- Rost B, Sander C (1994) Conservation and prediction of solvent accessibility in protein families. *Proteins* 20:216–226
- Rost B, Sander C, Schneider R (1994) PHD – an automatic mail server for protein secondary structure prediction. *Comput Appl Biosci* 10:53–60
- Ruan J, Wang K, Yang J, Kurgan LA, Cios KJ (2005) Highly accurate and consistent method for prediction of helix and strand content from primary protein sequences. *Artif Intell Med* 35:19–35
- Saeys Y, Inza I, Larrañaga P (2007) A review of feature selection techniques in bioinformatics. *Bioinformatics* 23(19):2507–2517
- Saha S, Raghava GP (2006) Prediction of continuous B-cell epitopes in an antigen using recurrent neural network. *Proteins* 65:40–48
- Sidhu A, Yang ZR (2006) Prediction of signal peptides using bio-basis function neural networks and decision trees. *Appl Bioinformatics* 5:13–19
- Vedani A, Dobler M (2000) Multi-dimensional QSAR in drug research. Predicting binding affinities, toxicity and pharmacokinetic parameters. *Prog Drug Res* 55:105–135
- Wilkinson DJ (2007) Bayesian methods in bioinformatics and computational systems biology. *Brief Bioinformatics* 8(2):109–116

- Xue B, Dor O, Faraggi E, Zhou Y (2008) Real-value prediction of backbone torsion angles. *Proteins* 72:427–433
- Yang ZR, Thomson R (2005) Bio-basis function neural network for prediction of protease cleavage sites in proteins. *IEEE Trans Neural Netw* 16:263–274
- Ye L, Liu T, Wu Z, Zhou R (2008) Sequence-based protein domain boundary prediction using BP neural network with various property profiles. *Proteins* 71:300–307
- Zhang GZ, Huang DS (2004) Prediction of inter-residue contacts map based on genetic algorithm optimized radial basis function neural network and binary input encoding scheme. *J Comput Aided Mol Des* 18:797–810
- Zhou HX, Shan Y (2001) Prediction of protein interaction sites from sequence profile and residue neighbor list. *Proteins* 44:336–343
- Zou L, Wang Z, Huang J (2007) Prediction of subcellular localization of eukaryotic proteins using position-specific profiles and neural network with weighted inputs. *J Genet Genomics* 34:1080–1087

