Neural Networks for Prediction of Nucleotide Sequences by using Genomic Signals

PAUL CRISTEA[1], VALERI MLADENOV[2], RODICA TUDUCE[1], GEORGI TSENOV[2], SIMONA PETRAKIEVA[2]

[1]Biomedical Engineering Center, Universitry "Politechnica" of Bucharest, Spl. Independentei 313, sect. 6, 060042 Bucharest, ROMANIA
e-mail: pcristea@dsp.pub.ro
[2]Department of Theoretical Electrical Engineering, Faculty of Automatics, Technical University of Sofia, 8, Kl. Ohridski St, Sofia-1000, BULGARIA
e-mail: valerim@tu-sofia.bg

Abstract: - The conversion of symbolic sequences into complex genomic signals allows using signal processing methods for the handling and analysis of nucleotide sequences. This methodology reveals surprizing regularities, both locally and at a global scale, allowing us to predict nucleotides in a sequence, when knowing the preceding ones. Such experiments have a major biologic significance, as they explore the possibility and the efficiency of error correction in processes like replication, transcription and translation.

Key-Words: - Genomic signals, Nucleotide Sequences, Time series prediction, Sequence prediction, Neural networks

1 Introduction

The conversion of symbolic sequences into complex genomic signals [3] allows using signal processing methods for the handling and analysis of nucleotide sequences. This methodology reveals surprising regularities, both locally and at a global scale, which would be difficult to identify by using only statistical analysis and pattern matching, as currently done for symbolic sequences. The approach is useful for studying large scale features of chromosomes [5, 10, 11], detect mutations in small and medium genomes such as those of pathogens [4, 6, 7, 8, 9] and model some ribosome functional properties [5, 14].

The regularities in the distribution of nucleotides [1] and pairs of nucleotides, reflected in the low values and predictable variation of the nucleotide imbalance (cumulated phase) and nucleotide pair imbalance (unwrapped phase), show that a genome has a multi-level ordered structure, despite the gene low compressibility. Mutations, such as those in the genomic signals of related pathogen strains, tend to mutually compensate, so that overall regularities of chromosomes are conserved. A direct consequence of this statistical regularity is that SNPs appear rarely isolated and more often in correlated groups, sometimes placed at large distances along the nucleotide sequences [3].

On the other hand, the regularity of genomic signals allows using techniques similar to those used in time series prediction [15] to estimate the nucleotides in a sequence when knowing the preceding ones [5,14]. Such experiments have a biologic significance, as they explore the possibility and the efficiency of error correction in processes like replication, transcription and translation.

In previous work, the efficiency of the prediction has been improved by using a two step procedure which comprises a principal analysis step (PCA) and a feed-forward ANN [13]. The PCA retains only the high variance components of the input signal, which are used by the ANN to perform the actual prediction.

The present paper uses either directly feed forward neural networks, or preceding step for feature extraction. This improves the prediction accuracy and increases the efficiency of the system. The results of experiments with this prediction model that show a remarkable low error rate. This behavior is the direct effect of the regularities in the structure of the genomic sequences [1, 3, 11].

2 Nucleotide Representation

The mapping we are using [3] is a one-to-one (bijective) unbiased representation of nucleotide equivalence classes, which attaches complex numbers to adenine, cytosine, guanine and thymine nucleotides:

$$A = 1 + j, \quad C = -1 - j, \quad G = -1 + j, \quad T = 1 - j. \quad (1)$$

While conserving the information in the initial symbolic sequence, this mapping introduces no artifacts related to specific assumptions on the types of interaction that characterize the nucleotides.

Correspondingly, the distribution of nucleotides along a sequence is described by the *nucleotide imbalance*:

$$N = 3(n_G - n_C) + (n_A - n_T), \quad (2)$$

where n_A, n_C, n_G and n_T are the numbers of adenine, cytosine, guanine and thymine nucleotides in the sequence, from the first to the current entry, while the distribution of nucleotide pairs by the *nucleotide pair imbalance*:

$$P = n_+ - n_-, \quad (3)$$

where n_+ is the number of positive pairs (A→G, G→C, C→T, T→A), and n_- the number of negative pairs (A→T, T→C, C→G, G→A).

The genomic signal approach reveals large scale features of DNA sequences that are maintained over distances of 10^6 – 10^8 base pairs, including both coding and non-coding regions [3,5,10,11]. The methodology is also adequate for the study of pathogen variability and the identification of multiple drug resistance, important for fast diagnoses and prompt socio-medical decisions in contamination with pathogens such as *Human immunodeficiency virus* (*HIV*) [9], *Avian influenza virus* (*H5N1*) [8] and *Mycobacterium tuberculosis* (*MT*) [2,4,6].

3 Protease Gene of *HIV-1* Clade F

The genomic signal representation has been used for the analysis and nucleotide prediction study of the protease (*PR*) gene of the Human immunodeficiency virus type 1 (*HIV-1*) clade F isolated from Romanian patients. The *PR* gene is an essential functional gene of the *HIV-1* virus, which encodes the two identical peptide chains in the structure of the protease enzyme. The *PR* gene consists of 297 base pairs, located along the 1799...2095 bp segment of the NC001802 sequence [16].

The *PR* enzyme is a small dimension dimmer that plays a vital role in the *HIV* life cycle by chopping up the long strands of polyprotein synthesized by the *HIV* RNA. After being generated by using the human ribosome in the attacked cell, the polyprotein molecules are cut by *PR* in smaller pieces corresponding to the viral proteins, at the proper sites and proper timing. The two *PR* enzyme chains, each 99 amino acids long, form a tunnel that holds the polyprotein, which is cut by the enzyme active site located in the center of the tunnel. Because of its importance, protease has been selected as a preferred target for anti-*HIV* drugs. The protease inhibitor drugs bind to PR to block its action.

4 Genomic Signals of *PR* Gene *vs.* Global Representation

The *PR* gene of *HIV-1* subtype *F* virions isolated from Romanian patients have been sequenced at the National Institute of Infectious Diseases "Prof.Dr.Matei Bals", Bucharest.

The symbolic sequences of the *PR* gene for 30 patients have been converted into complex genomic signals using (1). The nucleotide imbalance *N* (cumulated phase $\theta_c = \pi N / 4$) and the nucleotide pair imbalance *P* (unwrapped phase $\theta_u = \pi P / 2$) have been computed for the genomic signals using (2) and (3), respectively.

The signals have been classified in three groups taking into account the clinical behavior of the patients with respect to the response to current antiretroviral treatment:

• Sensitive (S) - the patient responds to the antiretro-viral treatment (wild type pathogen),

• Resistant (R) - patient resistant to one anti-*HIV* drug.

• Multiresistant (M) - patient resistant to several anti-*HIV* drugs.

The data set that we have used in this work contains equal groups of each type (10 patients who are sensitive, 10 patients who are resistant and 10 patients who are multiresistant).

Fig.1 shows the nucleotide imbalance N (cumulated phase) and the nucleotide pair imbalance P (unwrapped phase), for the sensitive, resistant and multiresistant patients.

The data have also been analyzed directly from the point of view of the nucleotide classes content. Figures 2-4 show the relative single and double nucleotide class content of the 30 individual sets of data, using the mapping given in Table 1.

Class	IUPAC symbol	Complex representation	Scalar label
Adenine	A	[1, 1]	1
Guanine	G	[1,-1]	2
Cytosine	C	[-1,-1]	3
Thymine	T	[-1,1]	4
Week bond	W	[1,0]	5
Purines	R	[0,1]	6
Strong bond	S	[-1,0]	7
Pyrimidines	Y	[0,-1]	8

Table 1 The data set mapping used for data presentation.

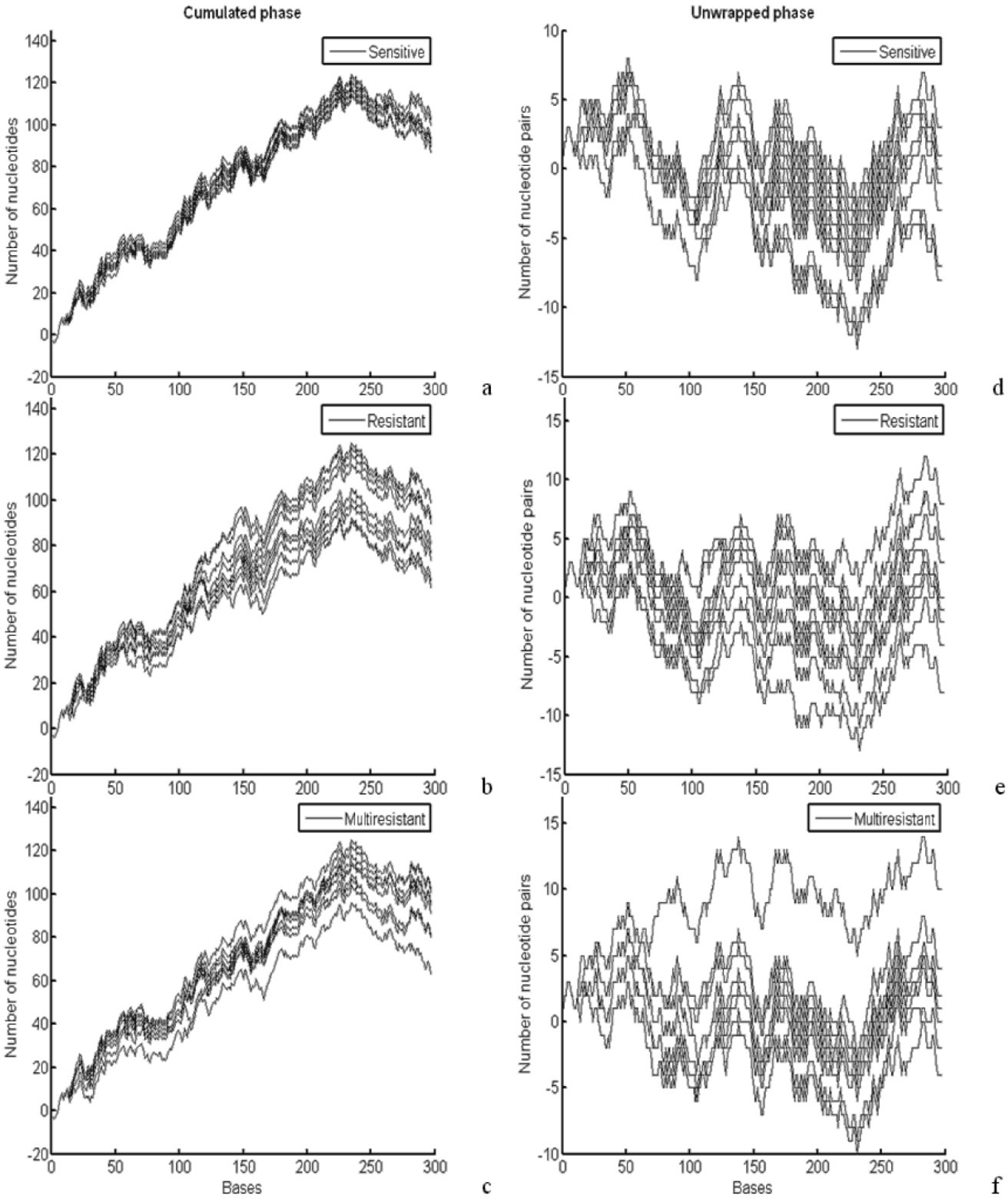

Fig.1 Nucleotide imbalance *N* (cumulated phase) and the nucleotide pair imbalance *P* (unwrapped phase) of the *PR* gene genomic signals, for sensitive, resistant and multiresistant patients.

The diagrams in Figures 2, 3 and 4 show that the nucleotide class content of the data for patients belonging to the same group of resistance to drugs is almost the same, as expected. Additionally, Fig. 5 shows that the average values of the nucleotide class content for different sets of data is also almost the same. This proves the need for an adequate description and analysis of genetic information. The global statistical analysis of the nucleotide class content of the gene data does not detect the specific changes resulting in drug resistance. As shown in previous work [6, 7, 8, 9], the development of the

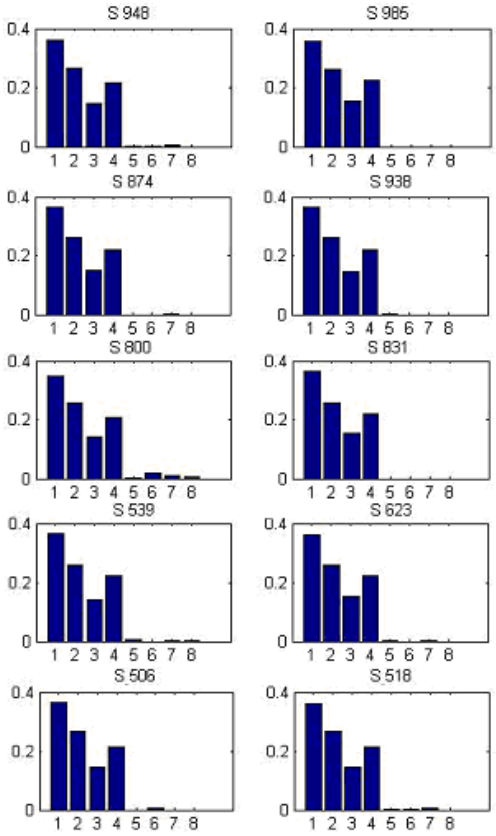

Fig.2 Relative nucleotide class content of *PR* genes isolated from treatment sensitive (S) patients.

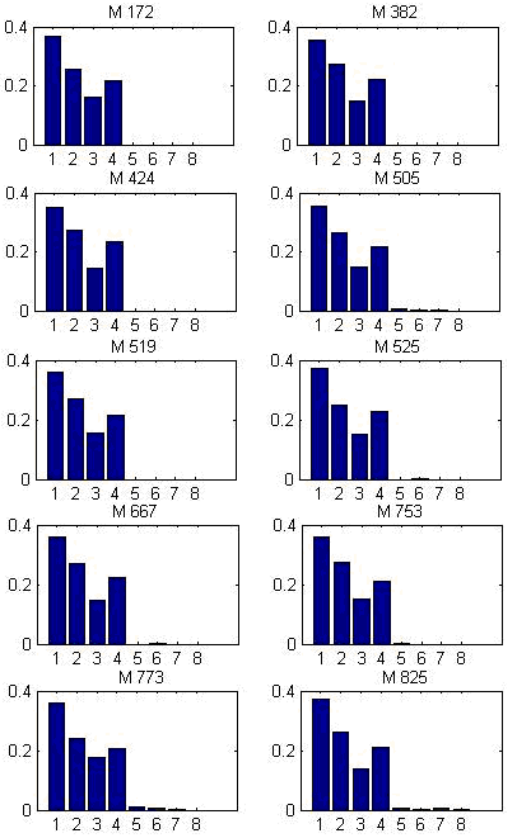

Fig.4 Relative nucleotide class content of *PR* genes isolated from treatment multiple-resistant (M) patients.

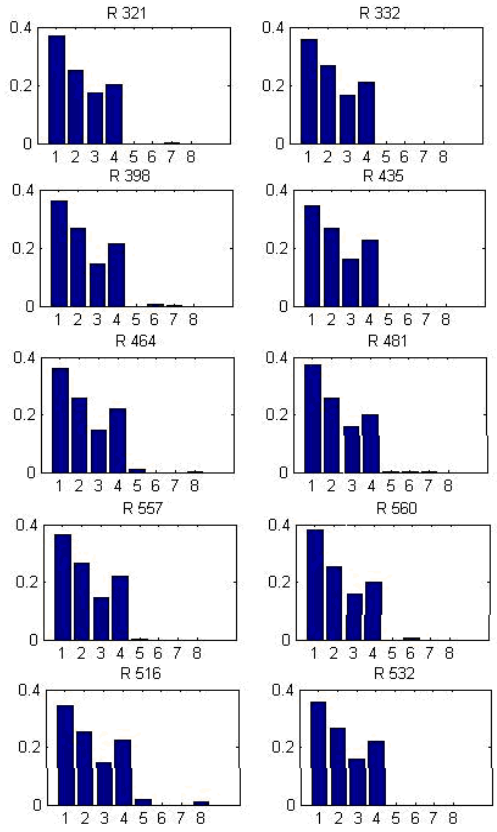

Fig.3 Relative nucleotide class content of PR genes isolated from treatment resistant (R) patients.

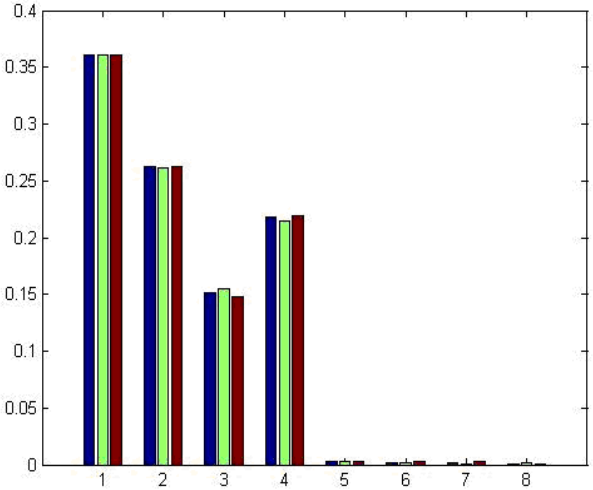

Fig.5 Average values of the nucleotide class content of the three types of drug resistant patients (S, R, M).

phenotypic resistance to drugs, at clinical scale, can be consistently linked to mutations in certain loci of the gene involved, at molecular scale. The specific locations of the mutations determine the drug resistance, while the nucleotide class content of the genes for various groups of drug resistant patients, and their average values do not.

The negative result of this investigation, performed at the level of global statistical features,

confirms the previous work that clearly linked pathogen resistance to treatment, on one side, to the detailed structure of the nucleotide sequences of characteristic genes, on the other side. The specificity of the genomic signals prompted us to further develop a new approach to the nucleotide prediction experiments, complementary to the work reported in [5].

When we work with GeneBank data, there are only the four main bases A,T,C,G. This is because in the DB is shown the most probable (common) strain and they mention the possible variations (SNPs). When we worked with the real experimental data, we observed that we have some co-existing strains in the biological samples. Consequently, if there are several bases in certain positions, the analysis will give the multi-nucleotide classes:

W = (A, T), Y = (T, C), S = (C, G), R = (G, A), K = (G, T), M = (A, C), B = nonA= (C, G, T), V = nonT = (A, C, G), D = nonC = (A, G, T) H = nonG = (A, C, T), and N = (A, C, G, T).

When processing the results, we can either successively replace a class with its components, or devise adequate software tools able to operate simultaneously with all the possible versions.

5 Prediction of Nucleotide Sequences

As shown in [5], the prediction of nucleotide sequences in a way similar to time series prediction is also an investigation of the theoretical possibility of ribosomes to use the redundancy in the genomic sequences to correct errors when synthesizing polypeptide chains based on the information brought by mRNA. The same question can be formulated for DNA replication and DNA transcription to mRNA.

The description of the ribosomal machine holds that a human ribosome maintains the contact with the previous 35 nucleotides and with the next 3 nucleotides, when processing the current nucleotide. In order to avoid contradictions and conceptual difficulties, the model we adopted for checking the possibility to predict nucleotide sequences uses only the information from the previous nucleotides. The previous 35 samples of the nucleotide sequence, converted into complex numbers, *i.e.*, 70 real and imaginary components, are applied at the input of the system, which is trained to produce an estimate of the current nucleotide.

The input was either applied directly to the NN input, or after preprocessing step to extract essential features, with a significant increase in system performance.

5.1 ANN Sequence Prediction

In order to find out the best NN structure suited to predict the described nucleotide sequences we created a large variety of NN types in MATLAB©, applying different training algorithms to every type in the training phase. We also experimented with the layer's activation functions, the number of neurons in the hidden layer and the number of input neurons (here we have two types of systems – with 70 input neurons or with compact conversion with 35 input neurons, when converting the gene data from complex form to some real number form).

The dataset for each patient consists of 297 samples and overall the total data set consists of 7830 instances composing the supervised training set, i.e., pairs of input vectors and target outputs. The dataset feeded to the NN is formed as we take every 35 sample sequence as input element and the 36-th as output element (desired output). The next element in the dataset is formed in the same fashion, with the difference that we move the window by one sample. In this case for every element in the dataset the second sample from the previous element becomes the first sample and the sample used as desired output on the previous dataset element becomes the 35-th final sample and the sample after it becomes the target element. For instance, samples 1…35 of some patient data set are used as an input and sample 36 as target, samples 2…36 used as input and 37 as target and etc. In such a fashion we process all the data from one of the patient groups, so in this case we have 297-36=261 input-target sets for one patient that can be obtained and 261*30=7830 gives the overall dataset number (where 30 is the number of patients).

The formed dataset has been randomly shuffled and divided then in three equally sized subsets (2610 elements for each): one third used for training the networks, one third for validation (early stopping to avoid overtraining) and one third is used for testing (for evaluation of network performance).

The first subset is the training set, which is used for computing the gradient and updating the network weights and biases. The second subset is the validation set. The error on the validation set is monitored during the training process as the validation error will normally decrease during the initial phase of training, as does the training set error. However, when the network begins to over-fitting the data, the error on the validation set will typically begin to rise and when this happens for certain number of iterations the training is stopped, and the weights and biases are returned at the minimum of the validation error. The test set is not used during

the training, but it is used to compare the different models when simulation the trained networks wit it.

For every different network type we started a training algorithm in the end of which we measured the current network's performance. In this step we apply the testing dataset to the inputs of the trained NN and form a measure (in percents) for correct classification rate by subtracting the desired network outputs from the obtained network output results after simulation.

After trying a large variety of candidate systems, the MATLAB© NN toolbox has been used to create a feed-forward neural network with 70 input neurons, one hidden-layer with a variable number of neurons and two output neurons (as schematically shown in Fig.6), which provided the best results. For this network, the first 35 inputs consist of the real parts of the previous 35 samples of the complex genomic signal, while the last 35 inputs are the corresponding imaginary parts. The two outputs are the real and imaginary parts of the predicted sample. Also, the hidden neuron number has been varied to find the optimal network architecture. The neurons were provided with the MATLAB© *tansig* activation function.

This neural network has been trained by using the MATLAB© *trainrp* (resilient backpropagation) training algorithm for its computational and performance advantages.

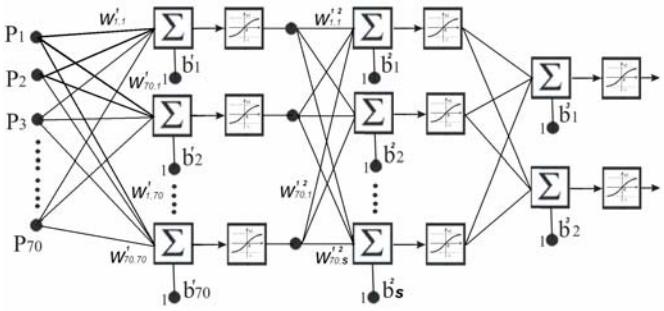

Fig.6. The Neural Network structure used for time-series prediction.

The Oja formula for the number of neurons in the hidden layer would recommend 8 neurons, but we got slightly better experimental results when using a larger number of neurons in the hidden layer.

The performance measured by varying the hidden layer neuron number to search for the best NN structure is given in Fig. 7.

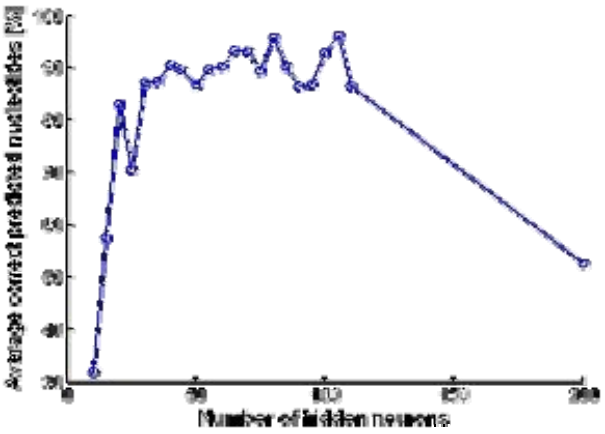

Fig.7. Average correct predicted nucleotides for various numbers of hidden neurons of the ANN.

A surprisingly high prediction accuracy of 96% has been obtained for a network with 105 neurons in the hidden layer. The marked decrease for large number of hidden neurons is caused perhaps by overfitting.

5.2 Feature Extraction

In this approach the prediction problem is converted into a classification problem. The extraction of relevant features allows us to make a code prediction on basis of the extracted information. Theoretically, it could be possible to have certain statistical common properties of various data sets that would facilitate code prediction. However, because of the large variability of the code, it is necessary to perform some form of feature extraction to overcome this variability or stick to the classic forecasting solutions as shown previously in 5.1. Obtaining a feature set space can simplify the classification task. Relevant features lead to accurate classification.

The prediction task has been simplified by reducing the set of input sample values. The double nucleotide classes have been replaced by single nucleotide ones in the input and, correspondingly, in the output (prediction) data. Specifically, the reduction replacements W $\rightarrow$ A, Y $\rightarrow$ T, S $\rightarrow$ C and R $\rightarrow$ G has been operated. Taking into account the low frequency of double nucleotide classes in the current sequencing data, the effect of this reduction is marginal.

To reduce the dimension of the input vectors, the complex representation has also been replaced for the prediction task with a real number coding such as A $\rightarrow$ -3, T $\rightarrow$ -1, C $\rightarrow$ 3, G $\rightarrow$ 1.

Using features such as number of the zero crossings, slope sign changes, distance between

vectors, mean, sum of squared elements, etc, we form a feature matrix in the way described as in the following example.

Data from 5 patients in each patient group are used for training and data from the remaining 5 patients are used for testing. Again, 35 data points represent the input vector, while the 36th sample is target value. In this case we use overall 3915 element dataset for training and the same number for testing the networks performance. For example for every nucleotide A, T, C and G we form two arrays:

- the network input matrix (if we have 80 occurrences of A, we shall have a 80x35 matrix divided into two 40x35 matrices – one for training and one for testing).
- the target vector (for these 80 occurrences of A, we shall have a vector of 80 '-3' elements, and for the current coding we divide it in two 40 element vectors for training and testing).

After applying the feature extraction for this example when using seven feature types we form a training and testing matrix with dimensions 40x7. In this case every row contains one specific feature value for the current 35 nucleotide sequence, and the next column holds the information for the next.

Only seven different kinds of features such as mean value of a data window

$$MAV = \frac{1}{N}\sum_{k=1}^{N}\left|\text{datawindow}_k\right|;$$

norm - the largest singular value in a data window, overall number of zero crossings in data window, the slope sign changes (which is the gradient of the zero crossings feature) in a data window, median - is the median value of the elements in datawindow, distance between the obtained vectors and sum of the squares in a data window are used. When these features are extracted and a feature paramatrix is created the values for them for every dataset element are feeded to the inputs of the chosen neural network, as they appeared to be sufficient for the task. For every different patient type a distinct network was created and trained with train/test data for its specific class with the same network structure as the one shown in Fig.8. So, with the usage of these features a feed-forward neural network with seven neurons in the first layer, 3 neurons in the hidden layer, and one output neuron has been used. The activation functions used are MATLAB© *tansig* for the hidden neurons in the second layer, and MATLAB© *pureline* for the output neuron in the third layer.

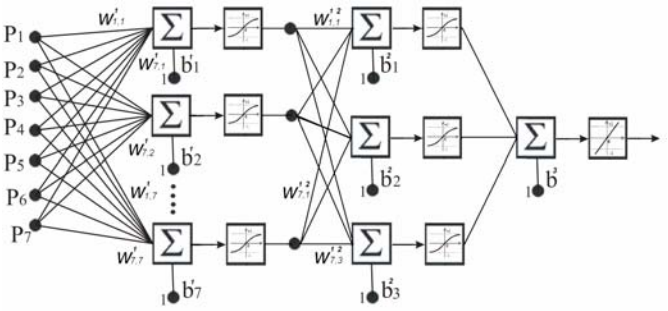

Fig.8. The Neural Network structure used.

The desired network outputs in this coding case are values like -3,-1,1 and 3. When testing the network we do not obtain strictly these values, but something similar (let's say instead of 3 we obtain 3.1 or 2.95). In order to have only -3,-1,1 and 3 as class number at the output, when we obtain result at the NN output we round the values in some interval (the interval is predetermined to be [-1;1]). In this case as an example, when having 2.4 and 3.3 at the network output, both these values are coded as 3.

Training was conducted using the gene data from half of the patients (from patients 1-5), and then we tested the network performance with data taken from the other half (from patient 6-10).

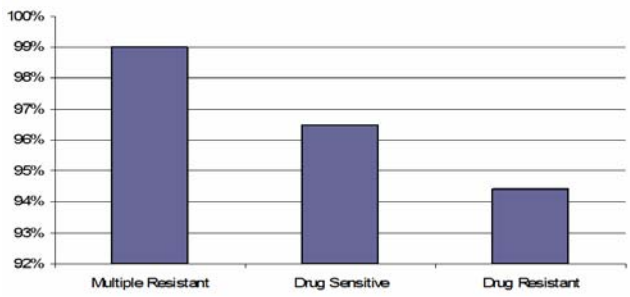

Fig.8. Correct prediction for the feature extraction approach.

As shown in Fig. 8, using the whole feature set we obtained a correct classification rate above 90% successful prediction for all the patient groups. Best classification results have been obtained for the Multiple Resistant group, while worst results was gained for the Drug Resistant group. Again, surprisingly high prediction accuracy has been obtained, as we did using ANN sequence prediction approach, and comparing both of them the feature extraction approach appeared to be better by 1%. Using different kind of features and performing PCA over the obtained feature matrix is an object of future work that holds the promise of improving even further this excellent result.

6 Conclusions

The correlation of genomic signals allows using techniques similar to those used in time series prediction [5, 16] to estimate the nucleotides in a sequence when knowing their preceding ones [5, 14]. Such experiments have a biologic significance, as they explore the possibility and the efficiency of error correction in processes like replication, transcription and translation.

The efficiency of the prediction is improved by using a two step procedure, comprising feature extraction and nucleotide prediction. The paper presents results of experiments with this prediction model that show a remarkable good efficiency. This behavior is the direct effect of the regularities and correlations in the structure of the genomic sequences.

Acknowledgment

The work was supported in part by grants from the Ministry of Education and Research of Romania, University "Politehnica" of Bucharest, the CEEX Program contract 50/2005 and CNCSIS 963-5/2007, the Ministry of Education and Science of Bulgaria, in the framework of the bilateral Scientific, Technological and Innovative Co-operation with Romania - project БРС-08, the Technical University - Sofia, Research Project No:08004ni-8/2008 and by the Franqui Research Project ADSI 133/2007, ETRO, Vrije Universiteit Brussel, Belgium.

References

[1] E.Chargaff, Structure and function of nucleic acids as cell constituents, Fed. Proc., 10, 1951, p. 654–659.
[2] S.T.Cole, R.Brosch, J.Parkhill, et al., Deciphering the biology of Mycobacterium tuberculosis from the complete genome sequence, Nature, 393 (6685) , 1998, p. 537-544.
[3] P.D.Cristea, Chapter 1, Representation and analysis of DNA sequences, in *Genomic Signal Processing and Statistics*, E. Daugherty, I. Shmulevich, J. Chen and Z.J. Wang, Eds., EURASIP Book Series on Signal Proc. & Comm., Hindawi Publ. Corp., 2005, pp. 15–65.
[4] P.D.Cristea, R.A.Tuduce, Analysis of inserts in prokaryote genomes, Proc. of SPIE, 6859-47, 2008.
[5] P.D.Cristea, Rodica Tuduce, I. Nastac, J. Cornelis, R. Deklerck, M. Andrei, Signal Representation and Processing of Nucleotide Sequences, Proc. of IEEE BIBE, Boston, MA, 2007, pp.1214-1219.
[6] P.D.Cristea, Genomic Signal Analysis of Mycobacterium tuberculosis, Proc. of SPIE, vol. 6447, 2007, pp. C-1 – C8.
[7] P.D.Cristea, Pathogen Variability. A Genomic Signal Approach, International Journal of Computers, Communications & Control Vol. I (3), 2006, pp. 25-32.
[8] P.D.Cristea, Genomic Signal Analysis of Pathogen Variability, Progress in Biomedical Optics and Imaging, Proc. of SPIE, vol. 6088, 2006, pp. P1-P12.
[9] P.D.Cristea, R.A.Tuduce, D. Otelea, Study of HIV Variability based on Genomic Signal Analysis of Protease and Reverse Transcriptase Genes, Proc.2005 IEEE EMB, 27th Ann. Conf., Shanghai, China, 2005.
[10] P.D. Cristea, Genomic Signals of Re-Oriented ORFs, EURASIP – Journal on Applied Signal Processing, Special Issue on Genomic Signal Processing, 2004, pp. 132–137.
[11] P.D.Cristea, Large Scale Features in DNA Genomic Signals, Signal Processing, Special Issue on Genomic Signal Processing, ELSEVIER, 83, 2003, pp. 871–888.
[12] H.Demuth, M.Beale, M.Hagan, Neural Network Toolbox 5, User's Guide, The Mathworks, 2007.
[13] M.T.Hagan, H.B.Demuth, M.H.Beale, Neural Network Design, Boston, MA: PWS Publishing, 1996.
[14] Nastac, P. Cristea, Neuro-Adaptive Forecasting for Nonstationary Sequences, Proc. IEEE-SOFA 2005, pp. 179-186.
[15] NIH - Natl Centre for Biotechnology Information, Natl Inst of Health, Natl Library of Medicine, (NCBI/GenBank), http://www.ncbi.nlm.nih.gov/, 2008.
[16] A.C. Tsakoumis, P. Fessas, V.M. Mladenov, N.E. Mastorakis, Application of Chaotic Time Series for Short-Term Load Prediction, WSEAS Trans. on Systems, 2 (3), 2003, pp. 517-523.