IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 5, Ver. I (Sept - Oct. 2016), PP 01-10
www.iosrjournals.org

Applying Back Propagation Algorithm for classification of fragile genome sequence

Medha Patel[1], Dr. Devarshi Mehta[2,] Dr. Patrick Patterson[3,] Dr. Rakesh Rawal[4]
[1](*Research Scholar, Gujarat Technological University Ahmadabad, India)*
[2](*Associate professor, GLSICT, Ahmadabad, India)*
[3](*Head and Professor, Texas Tech University, Texas, USA)*
[4](*Gujarat Cancer Research Institute, Ahmadabad, India)*

Abstract : *Most frequently occurring recurrent chromosomal translocation allied with all subtype of leukemia are available in Mitel Mann Data base. We have retrieved about 55 such genome sequence from TIC dB data base with 100% similarity score and got noncoding sequence of chromosome 9 and 22 as positive example of fragile site. Another 55 housekeeping genome sequence is taken for classification purpose. For content based analysis we have extracted 20 features of frequency density of mono nucleotide and dinucleotide. The network is designed by determining hyper parameters like number of hidden layer, hidden neurons and input features. First we took 20 input features and there after 16 for reducing number of free parameters (i.e. weight space). Network is also pruned for succeeding experiments. The training strategy was also exhaustively explored, based on literature study and trial and error heuristic methods to achieve more and more accuracy. Regularization is also employed by cross validation and early stopping. We have achieved 95% accuracy for training data and 70% to test data in first experiment. To avoid this over fitting at last we could achieve 93% over all accuracy and outlier detection, too. We could be able to show that dinucleotide frequency density is important statistical feature for classifying genome sequence. This classifier can show the probability of fragility to occur in genome sequence at very early stage so as to deal with the diesis at prognosis phase.*
Keywords: *back propagation, cancer classification, leukemia, non-coding sequence*

I. Introduction

Classifying non parametric data in high dimension with limited number of training samples of imbalanced data set is a challenging task. Artificial Neural Networks are natural classifiers having ability to recognize classes with highly nonlinear boundaries. They are efficient in solving complex problems due to characteristics such as robustness, fault tolerances, adaptive learning and massively parallel analysis capabilities, and for a biological system it may be employed as tool for data driven discovery. The gene classification problem is still active area of research because of the attributes of the genome data, high dimensionality and small sample size. Furthermore, the underlying data distribution is also unknown, so nonparametric methods must be used to solve such problems. Leukemia is a genetic disorder and chromosomal translocation is considered as primary cause for it. It is commonly accepted belief that cancer cells modify their transcriptional state during the prognosis phase of the diseases. [4] If we can find the mechanism to determine the high probability of fragility of chromosome, it becomes more helpful to prevent the diseases at prognosis phase.

In this study, we have built a classifier by content based analysis of genome sequence by applying Back propagation algorithm to multilayer perceptron. Network architecture is pruned and cross validation technique is also employed for regularization/ generalization. We have achieved 93.6% accuracy for over all data and 80% accuracy for test data. F score and correlation coefficient is also 0.9345 and 0.8740.

The overall conclusion is we can successfully employ ANN techniques to genome sequence for classifying and determining probability of diseases to occur in priory and also can find useful pattern from data for further research.

II. Background

2.1 Artificial Neural Network

ANN is an information processing system that has been developed as a generalization of the mathematical model of human cognition (ability to know). It consists of simple computational units called neurons that are highly interconnected and each connection has a strength that is expressed by a positive or negative number called weight. The connection of neurons is normally arranged in layers and executed in parallel. The connections are categorized as network topology. The size of the weight controls the influence that one neuron has on other, with a positive weight excite an element and negative weight inhibit. Over all the activation of an element is determined by a combination (summation) of excitatory and inhibitory influence it receives from its neighbors. The weights of the net are adaptable which store the experimental knowledge from

task example through a process of learning. The information is stored in the connections and distributed throughout, so the network can function as a memory of brain. The memory is content addressable, in the sense that the information may be recalled by providing partial or even erroneous input pattern. The information is stored in association of other stored data; hence it is adaptable.

2.2 Learning Process

When desired results are known for the sample training data, the training is done by supervised learning. In supervised learning initial weights are selected randomly for a sample input values, actual outputs are compared with target outputs, calculate the error and adjust the weights according to algorithm considered so that the actual output is closer to the target output in next run with smaller error; i.e. net converges. The modification of weight depends on learning rate. The larger the learning rate faster the training proceed, but may result in oscillation or non-convergence. Once trained, it can be applied to classify new input patterns and to establish relationship between inputs output pairs by function approximation/regression. For biological data we can classify the input genes of same expression levels, genes with tumor and healthy tissue etc. The perceptron, Feed Forward Neural Network with back propagation and Support Vector Machine are some examples of supervised learning algorithm applications. Learning of ANNs is accomplished by adjusting the weights of interconnection, according to the learning algorithm developed. The algorithms are developed according to well defined learning rules which simulate the learning methodology of brain's mathematical models.

Back propagation is similar to LMS algorithm and is based on gradient descent: weights are modified in the direction that corresponds to the negative gradient of error measure. For successful application of this method differentiable node activation function is required. The major advantage of back propagation over LMS and perceptron learning is in expressing how an error at a higher (outer) layer of a multilayer network can be propagated backwards to node at lower (inner) layers of the network. The architecture for back propagation learning is usually multi layer, feed forward (full connection between nodes in adjacent layers, no intra layer connection) with one or more hidden layer with nonlinear activation function. (Most commonly used functions are sigmoid function).

III. Related Work

Application of ANN in genomic and proteomic data set especially in cancer study [14] gives detailed analysis of benefits and shortcomings of using ANN. The paper describes the regularization techniques like weight decay, re sampling and early stopping, cross validation and k-fold cross validation with reference of practically used in certain papers. Recent applications in genomics and proteomics with comparison of ANN with other machine learning methods are also narrated. Paper shows some references of superiority of ANN based methods. In [22] the importance of cancer diagnosis and classification is emphasized with examples of class comparison, class prediction and class discovery. In [4] Shannon mathematical theory of communication i.e. Shannon entropy is used for cancer biomarker discovery. The same method is explained in [6] and we have also described in brief in future scope for vertical analysis of genome sequence near break point.

The flood of genome data can be analyzed effectively only if appropriate feature selection technique is known. In [31] feature selection techniques is discussed with its pros and cons. Unlike in dimensionality reduction the feature selection technique does not alter the original representations of the attributes and so preserve the original semantics and hence offering the advantage of interpretability by domain expert. We also have selected the features on the basis of this so as to analyze the effects of dinucleotide sequence by subject experts. The paper also gives important information about existing software available for feature selection methods. In paper [13] important conclusions related to my research is discussed. micro RNAs are small **non-coding** sequence responsible for chromosomal fragile sites. there is positive correlation between fragility and repeats. Density of non-coding miRNAs is significant for fragility. This helped me in analyzing noncoding regions near break points, content based. In [2] classification is achieved with significant lower prediction error by multiple classifier system. The feature selection was by wrapper approach and 10-fold cross validation technique is used. [37] is very good survey paper describe the comparison of conventional statistical based classification techniques with neural network techniques. It also provides the exhaustive survey of feed forward neural network based classification superiority over other machine learning methods too. This supports our decision of using ANN in our research.

In [5] three benchmark dataset for leukemia, colon and lymphoma cancer are used for systematic evaluating the performance of seven different feature selection methods and four different classifiers and developed an ensemble classifier. In paper [26] FF network is enhanced by evolutionary programming, which optimize the dimension of the network. Here they have used 4 neurons in single hidden layer for around 30-dimension input vector. In paper [21] also 2308 x 150 x 4 network is used.

The paper [33] is completely biology based with engineering flavor of mechanics. From this paper I have taken following conclusion on the basis of which my research is established. 1. The flexibility of DNA

molecule is sequence dependent property and varies globally on the overall base composition of the DNA molecule. 2. The local stiffness of DNA is strongly depending on the sequence content. In [8] frequency profiles (includes the number of times each possible combination of A T C G of the given N length word) of genome sequence is used as feature. In our research we have the argument that break in chromosome occur between two nucleotides, taken N=2 and frequency profile is used.

In [29] properties of DNA like bend ability, stiffness, stacking energy, duplex stability are dependent on composition/content of sequence. In this paper SVM is used, and length of string is taken of 1000 base pairs. The number of positive and negative sample data is taken equal to avoid imbalanced datasets.

In [16] the importance of non-coding DNA, particularly intronic DNA is established and concluded that studies of genetic variation can successfully discriminate and identify functional elements in noncoding regions. in [23] network architecture was 64 x 26 x 2 which determine approximately 40% of input neurons as hidden neurons. We have also tried this heuristically. [17]. In [3] the results obtained are in favor of multi-layer perceptron with back propagation algorithm over SVM based approach for imbalance data. They have implemented random sub-sampling validation, to estimate unbiased error rate instead of k-fold cross validation. Content based analysis of sequence is done here in [30] where frequencies of codon are computed. Network architecture is with one hidden layer and 30% of input node in hidden layer. The performance measure used is correlation coefficient. Paper [34] survey ANN and related pattern recognition techniques like GenCANS, GeneParser, GeneFinder, GENECLUSTER, SOTA etc. in [36] DNA sequence of several bacteria are classified by ANN and dinucleotide composition. Paper [27] describes the machine learning techniques used in bioinformatics.

IV. materials and methods

4.1 Data Acquisition

An extensive amount of information about all type of chromosome rearrangement in cancer is stored in **Mitelmann** data base of chromosome aberration in cancer, a public database available at the cancer genome anatomy project [19]. Mitellman contains in total 60907 cases of 803 different fusions and gives information related to the chromosome aberration and characteristics of tumor. Most frequently occurring recurrent chromosomal translocation allied with all subtype of leukemia were retrieved from it.

(http://egap.nci.nih.gov/chromosome/mitelman)

These fusion sequence of all translocation were collected and identified from **TICdb** data base, with breakpoint. TICdb (Translocation breakpoint in Cancer Database) is comprehensive collection of finely mapped translocation break point which describes the genomic location of 1225 breakpoints in human tumors.

(www.unav.es/genetica/TICdb) Now one part of the sequence is undertaken for **BLAT** study. 100% similarity score is matched in UCSC browser and found the original non-coding sequence before the break. Now this genome sequence is considered as an example of a sequence with highest probability of breaking which is most important cause considered for leukemia.

An unknown sequence can be annotated by two approaches. One by aligning the sequence with already known sequence from database that is homology search, but here we have tried to annotate un coded region of sequence also and so we go for another approach.

The second is composition based approach. This method uses N length word or N-mer as featured. This N-mers are taken by sliding window and frequency profiles (i.e. how often each word occurs in a given sequence is used as an input feature. There are sixteen such pairs of dinucleotide exists due to four single nucleotide A, C, G and T. We have taken the frequency density of occurrence of dinucleotides as the feature extracted from the genome sequence for analysis purpose. The 500 base pairs up string and 500 pairs down string from breakpoint is considered. This 16-dimension input vector is prepared for each sample data. We have taken sliding window of size 2 and occurrence of each dinucleotide in calculated and divided by total number of such pairs. As the features are between 0 and 1 there is no need to normalize the data base. The sample data and feature is shown in following figure.

No.	A%	C%	G%	T%	AA%	AC%	AG%	AT%	CA%	CC%	CG%	CT%	GA%	GC%	GG%	GT%	TA%	TC%	TG%	TT%	
1	0.3	0.203	0.204	0.293	0.118	0.051	0.064	0.066	0.061	0.06	0.014	0.068	0.062	0.043	0.053	0.046	0.059	0.049	0.073	0.112	0.95
2	0.297	0.203	0.208	0.292	0.115	0.052	0.065	0.065	0.061	0.06	0.014	0.068	0.062	0.042	0.055	0.048	0.059	0.049	0.073	0.111	0.95
3	0.301	0.203	0.203	0.293	0.118	0.052	0.064	0.067	0.062	0.06	0.013	0.068	0.062	0.042	0.053	0.046	0.059	0.049	0.072	0.112	0.95
4	0.301	0.203	0.203	0.293	0.118	0.052	0.064	0.067	0.062	0.06	0.013	0.068	0.062	0.042	0.053	0.046	0.059	0.049	0.072	0.112	0.95
5	0.3	0.201	0.236	0.263	0.106	0.052	0.072	0.07	0.067	0.049	0.016	0.069	0.068	0.048	0.073	0.046	0.058	0.052	0.075	0.078	0.95
6	0.251	0.199	0.208	0.342	0.091	0.046	0.052	0.062	0.049	0.043	0.016	0.091	0.059	0.041	0.053	0.054	0.052	0.069	0.087	0.134	0.05
7	0.275	0.219	0.218	0.288	0.01	0.05	0.066	0.059	0.063	0.065	0.015	0.076	0.06	0.048	0.058	0.052	0.052	0.056	0.078	0.101	0.05
8	0.305	0.183	0.228	0.284	0.109	0.046	0.07	0.079	0.063	0.043	0.015	0.062	0.069	0.042	0.069	0.048	0.064	0.052	0.074	0.094	0.05
9	0.224	0.223	0.302	0.251	0.053	0.048	0.08	0.042	0.061	0.054	0.039	0.069	0.077	0.065	0.094	0.066	0.033	0.056	0.089	0.073	0.05
10	0.248	0.247	0.273	0.232	0.071	0.045	0.09	0.042	0.083	0.071	0.023	0.07	0.061	0.067	0.089	0.056	0.032	0.064	0.071	0.064	0.05
11	0.299	0.201	0.236	0.264	0.106	0.052	0.072	0.069	0.067	0.049	0.016	0.069	0.069	0.048	0.073	0.046	0.057	0.052	0.075	0.079	0.95
12	0.237	0.204	0.231	0.328	0.066	0.043	0.061	0.066	0.051	0.045	0.012	0.096	0.062	0.049	0.066	0.054	0.058	0.067	0.091	0.112	0.95

We have obtained 55 such positive examples of BCR and ABL genes with breakpoints and same number of housekeeping genes sequence as negative examples. Thus the input to the network is 20-dimension and in other experiment 16-dimension vector of 110 such sample data.

The desired output is taken as 0.95 instead of 1 for positive example and 0.05 for negative example because we have taken sigmoid activation function for all neurons. For sigmoid function we get output as 0 only for net input to the neuron as $-\infty$ and 1 only for $+\infty$ input which is not possible. Since input to the neuron is $\sum x_i w_{ij}$ and x_i is finite, the weights required are infinitely high magnitude for training to converge. The magnitude of the first derivative of the sigmoid function is very small for large arguments, so weight updates are very small. Now this data is classified by using back propagation algorithm.

First and foremost, work is to check whether the decided features can actually discriminate the positive and negative examples or not. For that we have carried out a simple but statistically important method. We calculated the mean of positive and negative examples separately and plotted graphs for visual analysis. Once the validity of feature selected is checked, we concentrated on designing the network architecture and training strategy which is described in succeeding paragraphs. After validation of feature extracted, our next job was to design the network architecture. Here we deed the experiments with different hyper parameters like number of hidden nodes and input nodes. And for each such model the training parameter like learning rate and momentum are determined and network is trained accordingly.

4.2 Designing network architecture

Both the generalization and approximation ability of a feed forward neural network are closely related to the architecture of the network i.e. number of weights or free parameters in the network and the size of the training set. [3] The network we have designed is with through analysis and to some extent trial and error method so as to coop with the scarcity of labeled examples. First we have considered 4 single nucleotides and 16 di nucleotide as our input vector. Only one hidden layer is taken as the problem is of dichotomy and as per universal approximation theorem, this may suffice the requirement. In the hidden layer how many numbers of neuron should be taken, that is determined by exhaustive literature survey, heuristic, and trial and error method. The number of nodes must be large enough to form a decision region as complex as required by problem domain and at the same time not be excessively large so that the weights cannot be reliably estimated by available training set. [3] for 20-dimension input vector we have selected 8 neurons in hidden layers so free parameters became 20 x 8 + 8 x 1 + 9(bias) = 173 and available training size is 110. This may lead to over fitting and same is reflected in results. There after considering that break point is between two nucleotides we have reduced the input feature dimension and another 16 x 6 x 1 network is proposed. The free parameters are reduced and result to 16 x 6 + 6 x 1 + 7 = 108 and experiment is carried out with random presentation of inputs. At last after applying pruning techniques we have designed a 16 x 4 x 1 architecture.

4.3 SETTING THE PARAMETER VALUES AND TRAINING STRATEGY

Once an appropriate network is chosen one has to determine reasonable training strategy for specific problem domain. Here our input is frequency density of dinucleotide composition in sequence, there is no question to be some inputs to be very larger than others. So normalization is not required and at the same time initialization of weights for all connection are between -1 to +1 randomly. This is supported by sigmoid function.

The activation function selected is sigmoid as it is continuous monotonically increasing and its derivative is easy to compute.

Though per pattern weight update is more expensive than per epoch, we have selected per pattern weight updating. In fact, the two are more or less equivalent provided we maintain learning rate small. [3] There may be problem in learning the entire training set globally in per pattern presentation but it can be eliminated by random presentation of pattern i.e. the sequence of presentation of pattern should be shuffled from one epoch to the next. [3].

Weight vector changes in back propagation are proportional to negative gradient of error, but does not fix exact magnitude of the weight change. The change in weight space depends on the choice of learning rate eta η. All neurons in multi-layer perceptron should learn ideally at the same rate. [2] for a given neuron the learning rate should be inversely proportional to the square root of the connections made to that neuron. We took after exhaustive literature survey and trial and error method eta as 0.05 and momentum 0.1.

The learning rate eta in the BP algorithm with pattern update has to be kept small in order to maintain smooth trajectory of weight space. Large learning rate can lead to oscillation during learning. If the error surface (plotting MSE against network weight) is highly uneven and jagged with large number of local minima, there are much chances for a network to stuck in local minimum. This can be prevented by introducing a momentum alpha α in training. Momentum can accelerate the learning if weight update is in the same direction for two consecutive update and can prevent oscillations if weight update is in opposite direction for two consecutive update. It also helps in preventing network to stuck to some local minimum. A well-chosen alpha can significantly reduce the number of iteration for convergence. If the value of alpha is near to zero implies that past history has not much effect on weight update and for value near to one suggest that the current error has little effect in weight change. We selected value of alpha by study of literature and trial and error.

It is expected that the error i.e. the difference between desired and so we have decided the stopping criteria as either the sum error less than some specific value or number of iteration exceeds certain limits.

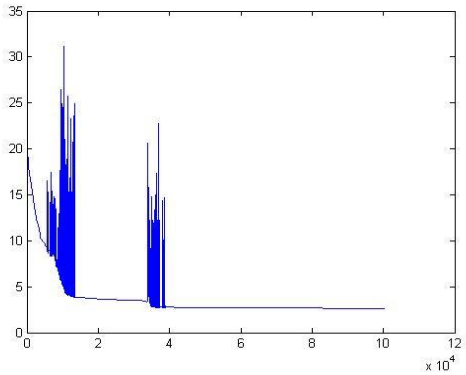
V. Result And discussion

The experimental results are presented along with its discussion here. First the graph of 4-nucleotide and 16-dinucleotide frequency density and its mean clearly show the difference in the mean for positive and negative samples for C, G and AA, AC, AT, GC, GG, TA and TT indicate that feature selected can be used for discrimination/classification.

5.1 Result and Error Graph

20 x 8 x 1 network

The classification is carried out with 20 x 8 x 1 network and after 100000 input presentations the sum error is 2.6198. Graph shows very less learning after 40000 iterations, which shows overfitting of data. The same thing is reflected in performance matrix also. The accuracy of data already seen in training is 95% and of test data it is 70%, which indicate poor generalization. This may be due to number of free parameters to adjust exceeds the sample data available. Here we have only 55 positive data and to avoid imbalance we have taken 55 negative data so, number of samples are fixed. So either we can reduce the input features or we can prune the network. We have applied both in next experiments

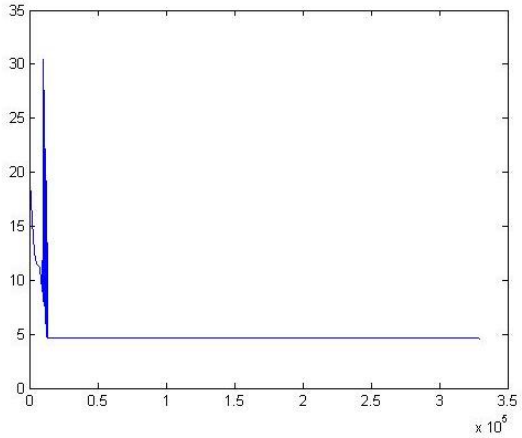


Graph 20*8*1 in training sum error graph for 100 data at a time 100175 steps

	TP	FN	FP	TN	Accuracy
Seen data	46	4	1	49	0.95
Test data	4	1	2	3	0.7

As per the logic and literature survey only dinucleotides may be considered for fragility of genome sequence, we have reduced the feature to 16 and designed another network model with 16 features and 8 hidden nodes.

16 x 8 x 1



Graph 16*8*1 in training sum error graph for 100 data at a time 329050 steps

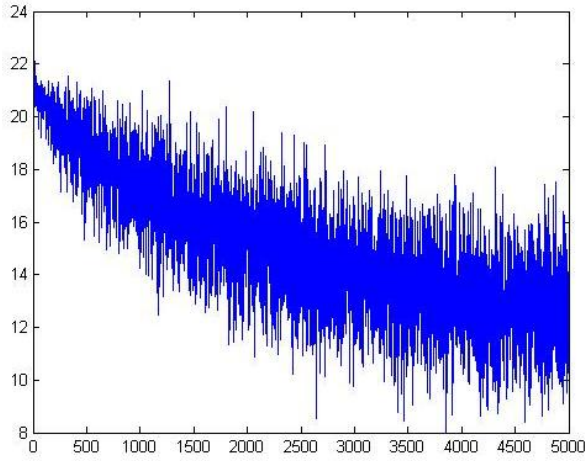
This network also gives the same accuracy result for training and test data which indirectly indicate that dropping out the single nucleotide frequency density feature is valid. So in further experiments we have concentrated only on pruning strategy.

	TP	FN	FP	TN	Accuracy
Seen data	46	4	1	49	0.95
Test data	4	1	2	3	0.7

Now in another experiment with 16x6x1 network model is carried out. Here we have selected three combination of learning rate and momentum.

16 x 6 x 1 (A)

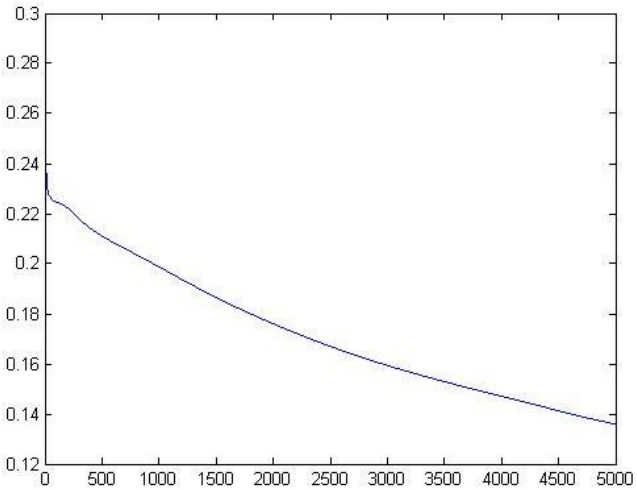
In first model of 16 x 6 x 1 the learning rate taken was 0.1 and momentum 0.5 and number ofiterations was 500000. The accuracy of seen data is 0.78 and of test data is 0.6. the graph of epoch error to epoch is shown below which is very oscillating so we have reduced learning rate and momentum in our next experiment and further 800000 iterations is carried out.



As the value of learning rate and momentum is high the graph shows oscillations, but still converging.

	TP	FN	FP	TN	Accuracy
Seen data	38	12	10	40	0.78
Test data	5	0	4	1	0.6

16 x 6 x 1 (B)



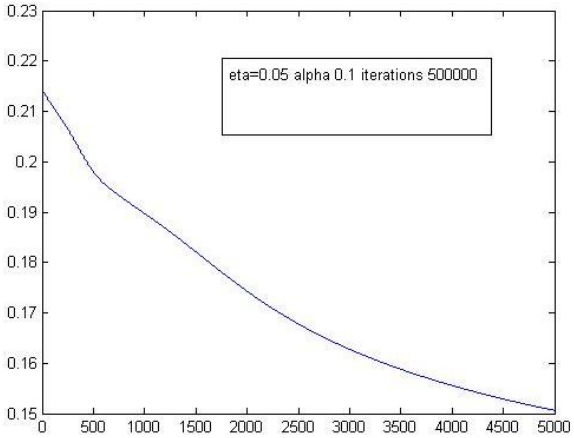
The result of accuracy is

	TP	FN	FP	TN	Accuracy
Seen data	37	13	5	45	0.82
Test data	5	0	0	5	1

The graph shows convergence and smoothness too for lower values of learning rate and momentum. The inclination of graph at end determines that network is still able to learn from sample data and the same is reflected in performance matrix. As test data is selected at random there may be a chance to achieve 100% accuracy for it. Epoch error is reduced to 0.136.

16 x 6 x 1 (C)

Here we have taken learning rate 0.05 and momentum as 0.1 with 500000 iterations. Epoch error is reduced below 0.15 and convergence indicates that network has still potential to learn.



The result of accuracy is

	TP	FN	FP	TN	Accuracy
Seen data	31	19	6	44	0.75
Test data	5	0	1	4	0.9

Performance measure shows that network is trained very slowly but slope at last shows that it is very convergent too. The result shows that the model is more generalized for unseen data. The experimental design is based on randomization and replication principles. We have fully explored replication and achieved reasonably good accuracy for seen and test data. Now as number of samples and input features are fixed we have employed 10 fold cross validation techniques. The graph of sum error at each round is shown. Here in table the result of all 110 sample data is shown. We can see that after 5^{th} round the network is reached at almost saturation position. It is not further modified by training. In a test data, 2^{nd} and 6^{th} sequence is consistently falsely predicted, so we can consider it as outliers.

Round	TP	FN	TN	FP	TP+TN	FP+FN	Accuracy = (TP+TN)/110	Error = (FP+FN)/110	Sensitivity = TP/55	Specificity = TN/55
1	50	5	53	2	103	7	0.936	0.064	0.909	0.964
2	35	20	46	9	81	29	0.736	0.264	0.636	0.836
3	43	12	35	20	78	32	0.709	0.291	0.782	0.636
4	41	14	49	6	90	20	0.818	0.182	0.745	0.891
5	48	7	53	2	101	9	0.918	0.082	0.873	0.964
6	50	5	51	4	101	9	0.918	0.082	0.909	0.927
7	50	5	51	4	101	9	0.918	0.082	0.909	0.927
8	50	5	53	2	103	7	0.936	0.064	0.909	0.964
9	50	5	53	2	103	7	0.936	0.064	0.909	0.964
10	50	5	53	2	103	7	0.936	0.064	0.909	0.964

The F-score after round 8 is

$$F = \frac{2 \text{ x precision x recall}}{\text{precision} + \text{recall}}$$

Where precision = TP/(TP + FP) and

recall = TP/(TP+FN)

$$F = \frac{2 \text{ x } 0.9615 \text{ x } 0.9090}{0.9615 + 0.9090}$$

$$F = \frac{1.748007}{1.8705}$$

$$F = 0.9345$$

The measure of correlation coefficient (MCC) is

$$MCC = \frac{(TP \text{ x } TN) - (FP \text{ x } FN)}{\sqrt{(TP + FP) \text{ x } (TP + FN) \text{ x } (TN + FP) \text{ x } (TN + FN)}}$$

$$MCC = \frac{(50 \text{ x } 53) - (5 \text{ x } 2)}{\sqrt{(50 + 2) \text{ x } (50 + 5) \text{ x } (53 + 2) \text{ x } (53 + 5)}}$$

$$MCC = \frac{2650 - 10}{\sqrt{52 \text{ x } 55 \text{ x } 55 \text{ x } 58}}$$

$$MCC = \frac{2640}{3020.50} \qquad MCC = 0.8740$$

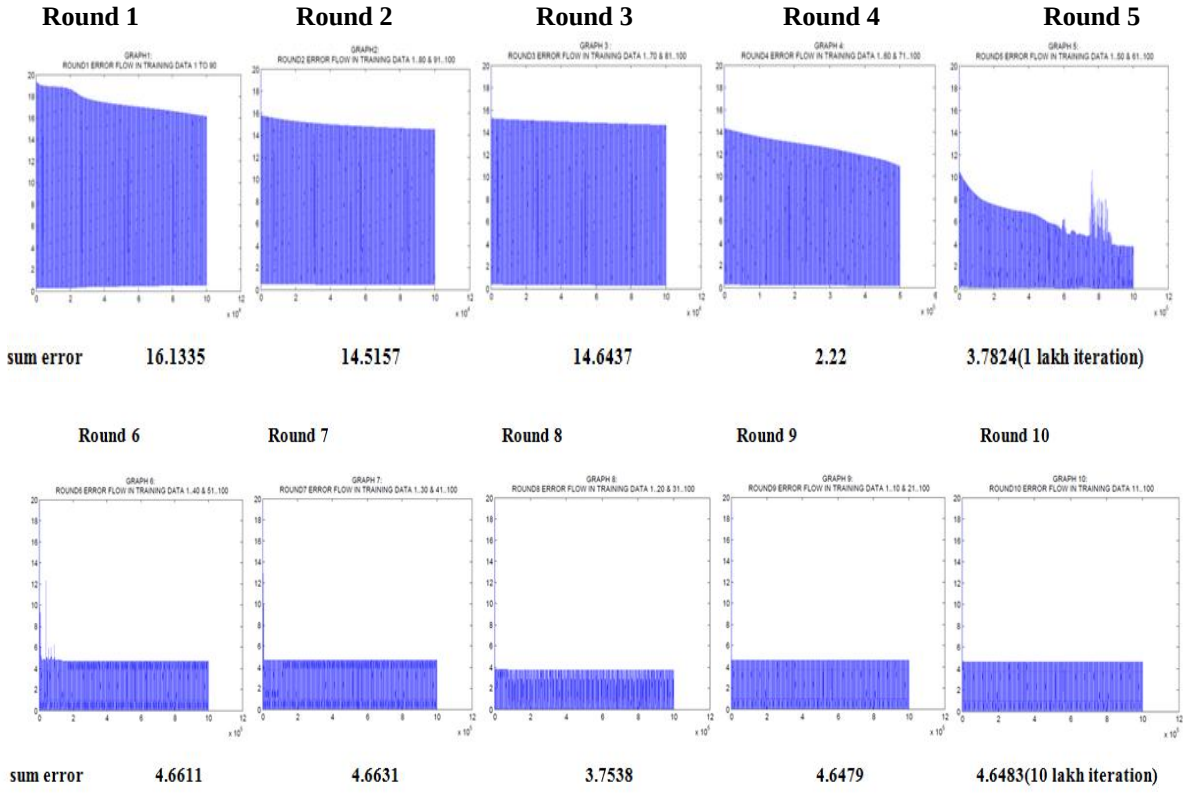
Over all the accuracy of classifier is 0.92 after round 8. The overall result of all experiment is as under.

	TP	TN	FP	FN	Accuracy	Remark
20 x 8 x 1	46	49	1	4	0.95	Over fitting
	4	3	2	1	0.70	
16 x 8 x 1	46	49	1	4	0.95	Over fitting
	4	3	2	1	0.70	
16 x 6 x 1(A)	37	45	5	13	0.82	May be a chance for test data
	5	5	0	0	1.00	
16 x 6 x 1(B)	38	40	10	12	0.78	Graph is oscillating
	5	1	4	0	0.60	
16 x 6 x 1(C)	31	44	5	19	0.75	Learning is incomplete
	5	4	1	0	0.90	
16 x 4 x 1	50	53	2	5	0.93	Outliers detection
Cross validation	4	4	1	1	0.80	

VI. Conclusion

However, by interdisciplinary research we may be able to find features from data which are meaningful biologically by analyzing weight space. In addition to that in future we may analyze the genome sequence vertically by Shannon entropy calculation. Further classifying with biological features like staking energy or melting temperature of genome sequence is also possible. Thus we can see that the network can be trained to classify the higher probability of fragility in original genome sequence with reasonable accuracy level. F-score also indicate balance between precision and recall. The classifier can be deployed for test purpose of detecting leukemia at prognosis phase for preventive therapy. The weight space obtained can be studied for biological significance of dinucleotides for data driven discovery.

16 x 4 x 1 (cross validation) Graph 1 to 10 for every round (10 round)



References

Journal

[1]. Albano, F., Anelli, L., Zagaria, A., Coccaro, N., D'Addabbo, P., Liso, V., Rocchi, M. & Specchia, G. (2010). Genomic segmental duplications on the basis of the t (9; 22) rearrangement in chronic myeloid leukemia. Oncogene, 29(17), 2509-2516.

[2]. Aminzadeh, F., Shadgar, B., &Osareh, A. (20SS, 3(1), 11-20.

[3]. Basu, S., &Plewczynski, D. (2010). AMS 3.0: prediction of post-translational modifications. BMC bioinformatics, 11(1), 1.

[4]. Berretta, R., &Moscato, P. (2010). Cancer biomarker discovery: the entropic hallmark. PLoS One, 5(8), e12262.

[5]. Cho, S. B., & Won, H. H. (2003, January). Machine learning in DNA microarray analysis for cancer classification. In Proceedings of the First Asia-Pacific bioinformatics conference on Bioinformatics 2003-Volume 19 (pp. 189-198). Australian Computer Society, Inc..

[6]. Dietterich, T. G. (2002, August). Machine learning for sequential data: A review. In Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR) (pp. 15-30). Springer Berlin Heidelberg.

[7]. Fadiel, A., &Naftolin, F. (2003). Microarray applications and challenges: a vast array of possibilities. Int Arch Biosci, 1, 111-1121.

[8]. Garbarine, E., DePasquale, J., Gadia, V., Polikar, R., & Rosen, G. (2011). Information-theoretic approaches to SVM feature selection for metagenome read classification. Computational biology and chemistry, 35(3), 199-209.

[9]. Jena, R. K., Aqel, M. M., Srivastava, P., &Mahanti, P. K. (2009). Soft computing methodologies in bioinformatics. European Journal of Scientific Research, 26(2), 189-203.

[10]. Kotsiantis, S., Kanellopoulos, D., &Pintelas, P. (2006). Handling imbalanced datasets: A review. GESTS International Transactions on Computer Science and Engineering, 30(1), 25-36.

[11]. Kuksa, P., &Pavlovic, V. (2007, September). Fast kernel methods for SVM sequence classifiers. In International Workshop on Algorithms in Bioinformatics (pp. 228-239). Springer Berlin Heidelberg.

[12]. Kumari, B., &Swarnkar, T. (2011). Filter versus wrapper feature subset selection in large dimensionality micro array: A review.

[13]. Lagana, A., Russo, F., Sismeiro, C., Giugno, R., Pulvirenti, A., & Ferro, A. (2010). Variability in the incidence of miRNAs and genes in fragile sites and the role of repeats and CpG islands in the distribution of genetic material.PloS one, 5(6), e11166.

[14]. Lancashire, L. J., Lemetre, C., & Ball, G. R. (2009). An introduction to artificial neural networks in bioinformatics—application to complex microarray and mass spectrometry datasets in cancer studies. Briefings in bioinformatics, bbp012.

[15]. Lippmann, R. (1987). An introduction to computing with neural nets. IEEE Assp magazine, 4(2), 4-22.

[16]. Lomelin, D., Jorgenson, E., &Risch, N. (2010). Human genetic variation recognizes functional elements in noncoding sequence. Genome research,20(3), 311-319.

[17]. Narayanan, A., Keedwell, E. C., & Olsson, B. (2002). Artificial intelligence techniques for bioinformatics. Applied bioinformatics, 1, 191-222.

[18]. Noble, W. S. (2006). What is a support vector machine?. Nature biotechnology, 24(12), 1565-1567.

[19]. Novo, F. J., de Mendíbil, I. O., &Vizmanos, J. L. (2007). TICdb: a collection of gene-mapped translocation breakpoints in cancer. BMC genomics, 8(1), 1.

[20]. Oliveira, A. M., & Fletcher, J. A. (2002). Translocation breakpoints in cancer.eLS.

[21]. Pal, N. R., Aguan, K., Sharma, A., & Amari, S. I. (2007). Discovering biomarkers from gene expression data for predicting cancer subgroups using neural networks and relational fuzzy clustering. BMC bioinformatics, 8(1), 1.

[22]. Perez-Diez, A., Morgun, A., &Shulzhenko, N. (2007). Microarrays for cancer diagnosis and classification. In Microarray Technology and Cancer Gene Profiling (pp. 74-85). Springer New York.
[23]. Peterson, L. E., & Coleman, M. A. (2005, September). Comparison of gene identification based on artificial neural network pre-processing with k-means cluster and principal component analysis. In International Workshop on Fuzzy Logic and Applications (pp. 267-276). Springer Berlin Heidelberg.
[24]. Peterson, L. E., & Coleman, M. A. (2005, September). Comparison of gene identification based on artificial neural network pre-processing with k-means cluster and principal component analysis. In International Workshop on Fuzzy Logic and Applications (pp. 267-276). Springer Berlin Heidelberg.
[25]. Peterson, L. E., Ozen, M., Erdem, H., Amini, A., Gomez, L., Nelson, C. C., &Ittmann, M. (2005, November). Artificial neural network analysis of DNA microarray-based prostate cancer recurrence. In 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (pp. 1-8). IEEE.
[26]. Pradhan, M., Pattnaik, S., &Mittra, B. (2010). Effective classification technique by blending of PPCA and EP-enhanced supervised classifier: Classifies microarray gene expression data. American Journal of Scientific Research, 11, 60-71.
[27]. Prompramote, S., Chen, Y., & Chen, Y. P. P. (2005). Machine learning in bioinformatics. In Bioinformatics technologies (pp. 117-153). Springer Berlin Heidelberg.
[28]. Rangannan, V., & Bansal, M. (2007). Identification and annotation of promoter regions in microbial genome sequences on the basis of DNA stability. Journal of biosciences, 32(1), 851-862.
[29]. Rawal, K., &Ramaswamy, R. (2011). Genome-wide analysis of mobile genetic element insertion sites. Nucleic acids research, 39(16), 6864-6878.
[30]. Rebello, S., Maheshwari, U., &DSouza, R. V. (2011). Back propagation neural network method for predicting Lac gene structures in Streptococcus pyogenes M Group A Streptococcus strains. International Journal of Biotechnology and Molecular Biology Research, 2(4), 61-72.
[31]. Saeys, Y., Inza, I., &Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. bioinformatics, 23(19), 2507-2517.
[32]. Score, J., Calasanz, M. J., Ottman, O., Pane, F., Yeh, R. F., Sobrinho-Simões, M. A., ... &Wiemels, J. (2010). Analysis of genomic breakpoints in p190 and p210 BCR–ABL indicate distinct mechanisms of formation.Leukemia, 24(10), 1742-1750.
[33]. Travers, A. A., & Thompson, J. M. T. (2004). An introduction to the mechanics of DNA. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, 362(1820), 1265-1279.
[34]. Valafar, F. Neural Network Applications in Biological Sequencing. InProceedings of the 2003 International Conference on Mathematics and Engineering Techniques in Medicine and Biological Sciences (pp. 24-27).
[35]. Vert, J. P. (2005). Kernel methods in genomics and computational biology.
[36]. You, W., Wang, K., Li, H., Jia, Y., Wu, X., & Du, Y. (2009, December). Classification of DNA sequences basing on the dinucleotide compositions. InComputational Intelligence and Design, 2009. ISCID'09. Second International Symposium on (Vol. 2, pp. 390-394). IEEE.
[37]. Zhang, G. P. (2000). Neural networks for classification: a survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 30(4), 451-462.
[38]. Zhang, Y., & Rowley, J. D. (2006). Chromatin structural elements and chromosomal translocations in leukemia. DNA repair, 5(9), 1282-1297.

Books

[1]. Alpaydin, E. (2010). Introduction to Machine Learning. New Delhi: PHI Learning.
[2]. Haykins, S. (2008). Neural Networks A Comprehensive Foundation, second edition. PEARSON, Prentice Hall
[3]. Kumar, S. (2nd reprint 2013). Neural Networks, A Classroom Approach. new Delhi: Mc Graw Hill Education.
[4]. Pevzner, P. (2000). Computational molecular biology: an algorithmic approach. MIT press.
[5]. Mitra, S., Datta, S., Perkins, T., & Michailidis, G. (2008). Introduction to machine learning and bioinformatics. CRC Press.
[6]. Gopal, S., Haake, A., Jones R., Tymann P., (2010). Bioinformatics. Tata McGraw Hill.

Thesis/dissertation

[1]. Ranawana, R. (2007). Intelligent multi-classifier design methods for the classification of imbalanced data sets (Doctoral dissertation, Ph. D. dissertation.–Univ. of Oxford, Oxford, UK).