Korean J. Chem. Eng., 33(4), 1318-1324 (2016)
DOI: 10.1007/s11814-015-0255-z

pISSN: 0256-1115
eISSN: 1975-7220

Artificial neural networks as classification and diagnostic tools for lymph node-negative breast cancers

Satya Eswari J.† **and Neha Chandrakar**

National Institute of Technology Raipur, Raipur-492010, India
(*Received 7 April 2015 • accepted 26 November 2015*)

Abstract−Artificial neural networks (ANNs) can be used to develop a technique to classify lymph node negative breast cancer that is prone to distant metastases based on gene expression signatures. The neural network used is a multilayered feed forward network that employs back propagation algorithm. Once trained with DNA microarray-based gene expression profiles of genes that were predictive of distant metastasis recurrence of lymph node negative breast cancer, the ANNs became capable of correctly classifying all samples and recognizing the genes most appropriate to the classification. To test the ability of the trained ANN models in recognizing lymph node negative breast cancer, we analyzed additional idle samples that were not used beforehand for the training procedure and obtained the correctly classified result in the validation set. For more substantial result, bootstrapping of training and testing dataset was performed as external validation. This study illustrates the potential application of ANN for breast tumor diagnosis and the identification of candidate targets in patients for therapy.

Keywords: Artificial Neural Network, Breast Cancer Diagnosis, DNA Microarray-based Gene Expression Profile, Classification

INTRODUCTION

Exploration of gene sequences reveals that it restrains enormous information which is difficult to decipher due to its complexity. In this postgenomic era, gene-expression profiling by means of cDNA microarrays has led to the simultaneous scrutiny of unique gene patterns associated with cancer which acts as markers and has been used to classify tumors into clinically relevant subtype [1] with members sharing common features. Implementation of artificial intelligence methods has fuelled the area of extraction of biologically significant features in the gene sequences [2,3]. Evolutionary algorithms such as genetic algorithms and differential evolution are used for optimization and classification of biological systems [4-7]. Neural network technique has evolved as an efficient classifier and also as a programming methodology for optimization of systems [8-11]. For the evaluation of gene-expression data, neural networks have been meticulously tested for their capability to precisely distinguish among cancers belonging to several diagnostic categories [12].

The transformation of microarray studies into clinical practice is still difficult due to the lack of comparison and overlap of results obtained from each individual study. The absence of concordant genes in different studies could be associated with the differences in studying samples and genes coming out from the upstream or downstream of oncogenic and anti-oncogenic pathways [13,14]. Gene expression mainly depends upon physical and molecular factors. Control of expression is vital to allow a cell to produce the gene products when it needs them; in turn, this gives cells the flexibility to adapt to a variable environment, external signals, damage to the cell, etc.; if any of the gene product overexpresses, then mutations may happen. Cancer is one of the diseases which particularly occur because of gene expression. There are certain candidate genes that particularly overlap with few among the gene signature, which can be due to gene-gene and gene-environment interaction [15] presence of polymorphic variation and haplotypes [16] or certain environmental and lifestyle factors [17]. All these intricate biological and physical factors can contribute wholly or partially to tumor formation.

When a tumor is prone to metastasis, aggressive adjuvant therapy can be prescribed, which has led to significant decreases in cancer mortality rates. But for the majority of patients with intermediate-risk breast cancer, the traditional factors are not strongly predictive. Thus, about 70-80% of lymph node-negative patients may undergo adjuvant chemotherapy when it is actually not required [1]. An enduring challenge is to recognize prognostic markers that are more directly related to disease and that can more accurately predict the risk of metastasis in individual patients. There is a need for precise classification of an individual patient's threat of disease recurrence to make certain that she receives suitable therapy. In the recent years, an increasing number of disease markers have been identified through analysis of genome-wide expression profiles [18,19,20]. Currently, not many diagnostic tools are available for individual risk assessment of patients.

In this study, our aim was to develop a gene-expression-based methodology using artificial neural network (ANN), which can be applied to provide quantitative predictions on disease outcome for patients with lymph-node-negative breast cancer. This work is focused on providing an appropriate diagnostic tool using candi-

†To whom correspondence should be addressed.
E-mail: eswari_iit@yahoo.co.in

date genes that specifically indicate the risk of distant recurrence.

ARTIFICIAL NEURAL NETWORKS

Functioning of nerve cells in humans is used to develop a computational method called as artificial neural networks (ANNs). ANNs are computer-based algorithms which are modelled on the structure and behavior of neurons in the human brain and can be trained to recognize and categorize complex patterns. The ANN parameters are adjusted for pattern recognition through error minimization during learning from experience. They can be calibrated by any type of input data, such as gene-expression levels generated by cDNA microarrays, and the output can be grouped into any given number of classes [12]. ANNs have been applied in the diagnosis of several types of diseases with different input data [21].

In this study, the classification using neural network is based on supervised learning. In supervised learning, the network is provided with a correct answer (output) for every input pattern. Weights are determined to allow the network to produce answers as close as possible to the known correct answers, which is done by using back-propagation algorithm. Prevalent studies have been established for classification by the statisticians, but the network-based method emerging as a new area of research is viewed to provide better efficiency for classification problems. We have used neural networks because they have numerous applications in handling large databases [22,23], which includes biochemical engineering, biomedical science and bioinformatics [2,3,21], DNA sequence analysis and biological pattern recognition [24]. There are a number of reasons to incorporate prognostic markers with ANN, which has been thought to provide an efficient diagnostic methodology; First, prognostic signature belongs to many functional classes, which suggests that different paths could lead to disease progression, hence providing better means of detection. Second, it provides models of the molecular mechanisms underlying metastasis. Finally, network-based classification achieves higher accuracy in prediction, as ascertained by selecting markers from one data set and applying them to a second independent validation data set [23]. Neural networks are used here for classification of lymph node negative breast cancer genes based on the information content of microarray data [1] due to its ability of training easily and tractability to tackle higher amount of information with good generalization ability, resulting in cost-effective and flexible modelling of large data sets. This leads to accuracy in predictive deduction, with potential to support clinical decision making and makes knowledge propagation easier by providing explanation, i.e., cross validation.

1. System Design of Artificial Neural Networks

1-1. Input/Output Mapping

ANN is a computational design method developed to imitate the operations of the human brain using interconnected parallel nodes. The weights assigned to the interconnections are called as inter connection weights representing its strength and storing learned information acquired after training [2]. The acclimatizing nature of ANNs enables them to procure some of the neurological processing capabilities such as learning and depicting inference from experience. The widely used ANN model is a multi-layered feed-forward network (MFFN) with multilayered perceptron (MLP), mostly consisting of three consecutively arranged layers of processing units [26,27]. The nonlinear function (f) is used to provide mapping between an input (x) and an output (y) i.e., y=f(x). The function f is approximated between x and y during the training process [21]. Each of three layers have their own nodes: input, hidden and output layers. Weighted links are used to connect all the nodes in each layer, i.e. input layer to the hidden layer nodes and the hidden to the output layer nodes. Numerical processing is not carried out in the input layer; it is done by the hidden and output layer nodes, and hence they are termed as active nodes [2,3].

1-2. Network Learning and Processing Schema

The network achieves learning by means of training that makes it recognize patterns in the data and predict the output. Network learning is achieved via two pathways. In the primary pathway, forward transfer of input data takes place. An input from the training data set is applied to the input nodes that, after calculating the weighted sum of the inputs, serve to the active node which is then transformed into output using a nonlinear activation function such as the sigmoid function. The output of computed hidden layer nodes now turns out to be the input to the output layer nodes, and the output is evaluated similarly. In the secondary pathway, the squared residuals defined in terms of target and prediction outputs follow a reverse path for updating the network weights to minimize a suitable function. The weight updating procedure when repeated for all the data in the training database completes a single iteration [2]. The value of squared error as a function of number of nodes in the hidden layer provides an optimized value of hidden nodes [21]. Since the number of hidden nodes is an amendable structural constraint, an unnecessary hidden unit in the network architecture leads to an oversized network; therefore, it requires an optimal value. Network simulations are established by methodologically altering the number of hidden units to avoid overfitting of the network giving the most favorable network architecture with the smallest extent of error for the test data [2].

1-3. Primary Pathway: Data Propagation

The input to the network consists of n-dimensional vector x_n and a unit bias θ_j. The incoming data is processed by multiplying the input to the neuron in the hidden layer with a weight w_{ij}, the product obtained is summed and unit bias θ_j is added [21] to get the activation state S_j:

$$S_j = \sum_{i=1}^{n} w_{ij} x_i + \theta_j \tag{1}$$

where i=1, 2, 3...n corresponds to number of inputs and j=1, 2, 3...m corresponds to number of neurons in the hidden layer.

The S_j is transformed through sigmoid function and the output O_j is calculated as:

$$O_j = f(S_j) = \frac{1}{1+e^{-S_j}} \tag{2}$$

where f represents the differentiable and non-decreasing function. The output of the first hidden layer neuron acts as input for the next neuron in the hidden layer and the last layer provides the network's output.

1-4. Secondary Pathway: Training Algorithm

Neural network training faces the most crucial hindrance as it requires a set of weights such that the prediction error, which is defined as the difference between the networks predicted output and the desired output, is minimized. The iterative training makes the network acquainted with patterns in the data and an internal model is created to provide predictions for the new input condition. Initially randomized weights are generated which are consequently adjusted so as to minimize the objective function E(w). E(w) as the residual of an observed value is the difference between the observed value and the estimated value of the quantity of interest (for example, a sample mean) defined as the mean squared residual between the prediction outputs y_{ik} and the target outputs d_{ik} for all the input patterns:

$$E(w)=\frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{m}(d_{ik}-y_{ik})^2 \quad (3)$$

The task of E(w) minimization is accomplished by training the network using a gradient descent technique called as generalized delta rule. According to this rule, the error function δ_{ik} between the hidden layer neurons to the output layer neuron k is computed:

$$\delta_{ik}=(d_{ik}-y_{ik})f'(S_{ik}) \quad (4)$$

The error function δ_{ij} between input neuron to hidden neuron j can be calculated as:

$$\delta_{ij}=f'(S_{ij})\sum_{k=1}^{m}\delta_{ik}w_{jk} \quad (5)$$

The weight change Δw of output to hidden layer after p^{th} data presentation is given by:

$$\Delta w_{jk}(p)=\eta\delta_{ik}O_{ik}+\alpha\Delta w_{jk}(p-1) \quad (6)$$

where η is the learning rate and α is the momentum factor. The updated weights are given by

$$w_{jk}(p)=w_{jk}(p-1)+\Delta w_{jk}(p) \quad (7)$$

After obtaining updated weights, a new training example is randomly selected, and the procedure is repeated until the reasonable reduction of the objective function is achieved.

METHODS

1. Creating Database

The gene profile was taken from the prognostic signatures reported by Wang et al. [1], which identified prognostic gene signatures for estrogen receptor (ER) positive and ER negative patient. The accession number was obtained which was then collected in FASTA format from the National Center for Biotechnology Information (NCBI) nucleotide database (www.ncbi.nlm.nih.gov/). The database collected was used for computational trial. The length of the sequences varied, but the length chosen for analysis was 490. 40 gene signatures were selected for neural network classification including most relevant gene signatures as reported in Chou et al. [28]. Each sequence collected from the NCBI nucleotide database (Genbank) has its own accession number. For example, the sample sequence of Homo sapiens Golgi membrane protein GP73 has the NCBI accession number NM_016548.1, the sample sequence of Homo sapiens chromosome-associated polypeptide C (CAP-C) has the NCBI accession number NM_005496.1 and the sample sequence of Homo sapiens H4 histone family, member H (H4FH) has the NCBI accession number NM_003543.2. Additional information related to gene signature along with their representative coding used for the development of ANN methodologies is given Table 1.

2. Data Preparing

Data encoding plays an essential function in improving the network performance. Neural networks are capable of processing many diverse forms of data, presented to the network in a suitable format for evaluation. There are different classes of inputs that should be aptly distinguished by the neural network, which requires input data in an appropriate layout; hence data encoding methodology preprocesses the data for preparing training database. DNA sequences of Homo sapiens are made up of four bases, A, T, G and C. These four bases should be represented in the form of a numerical vector in order to train a neural network for sequence classification. The numerical values for encoding the input data sequences were 0, 0.5, 1, 1.5 for the bases A, T, G, C respectively, and to represent the output gene signature, values chosen for the neural network methodology are presented in Table 1.

3. Experimental Protocol

Training sessions were performed for each configuration with randomly generated different initial weight factors. The number of neurons in the hidden layer was determined based on observation. First, we increased the number of neurons, one by one from 1 to 10. Optimum value for number of neuron in the hidden layer was obtained by adjusting network parameters. The prediction dataset acted as a supervisor set and the output was obtained accordingly. The progression of the network performance was evaluated by examining E(w) and the percentage of correct classifications in the training and validation set. After training, the weights of the networks were used in the validation phase. The overall architecture for gene classification is shown in Fig. 1.

RESULTS AND DISCUSSION

The aim of this work is to present a neural network methodology that can precisely identify the risk of tumor recurrence so that patients can be grouped as low risk and high risk in case of lymph node negative breast cancer.

1. Neural Network Training Results

1-1. Number of Neurons in the Hidden Layer

The recognition ability of the network in the training set was observed. Performance increased very quickly with the number of neurons in the hidden layer up to four hidden neurons. The performance became stable thereafter, and, with more than four neuron performance declined slightly. In accordance with these results, we chose to work with four hidden neurons.

1-2. Number of Iterations

The E(w) between actual values and the values predicted by the network declined very rapidly from a high starting value to a minimum value (close to 0) and became stabilized in 7000 iterations in the training set. In the validation set, a similar variation was observed, with acceptable minimal value, but not much near to 0 as

Table 1. Output coding for gene signatures

Accession number	Gene symbol	Gene description	Output dode for classification
AA772093	NEURL	Neuralized homolog (Drosophila)	1
AF114012	TNFSF12-TNFSF13	Tumor necrosis factor (ligand) superfamily, member 12	2
AF114013	TNFSF13	Tumor necrosis factor (ligand) superfamily, member 13	3
AF123759	CLN8	Ceroid-lipofuscinosis, neuronal 8 (epilepsy, progressive with mental retardation)	4
AF220152	TACC2	Transforming, acidic coiled-coil containing protein 2	5
AI057637	ACACB	Acetyl-CoA carboxylase beta	6
AI493245	CD44	CD44 molecule (Indian blood group)	7
AK001280	HDGFRP3	Hepatoma-derived growth factor, related protein 3	8
AL136877	SMC4	Structural maintenance of chromosomes 4	9
AL523310	MAP4	Microtubule-associated protein 4	10
AV693985	ETV2	Ets variant 2	11
BC005978	KPNA2	Karyopherin alpha 2 (RAG cohort 1, importin alpha 1)	12
BC006325	GTSE1	G-2 and S-phase expressed 1	13
BF055311	NEFL	Neurofilament, light polypeptide	14
BF055474	PHF11	PHD finger protein 11	15
NM_001175	ARHGDIB	Rho GDP dissociation inhibitor (GDI) beta	16
NM_001394	DUSP4	Dual specificity phosphatase 4	17
NM_001958	EEF1A2	Eukaryotic translation elongation factor 1 alpha 2	18
NM_002710	PPP1CC	Protein phosphatase 1, catalytic subunit, gamma isozyme	19
NM_002803	PSMC2	Proteasome (prosome, macropain) 26S subunit, ATPase, 2	20
NM_003543	HIST1H4H	Histone cluster 1, H4h	21
NM_004111	FEN1	Flap structure-specific endonuclease 1	22
NM_004470	FKBP2	FK506 binding protein 2, 13 kDa	23
NM_005030	PLK1	Polo-like kinase 1	24
NM_006314	CNKSR1	Connector enhancer of kinase suppressor of Ras 1	25
NM_006720	ABLIM1	Actin binding LIM protein 1	26
NM_007192	SUPT16H	Suppressor of Ty 16 homolog (*S. cerevisiae*)	27
NM_013936	OR12D2	Olfactory receptor, family 12, subfamily D, member 2	28
NM_014109	ATAD2	ATPase family, AAA domain containing 2	29
NM_014796	KIAA0748	KIAA0748	30
NM_015997	Clorf66	Chromosome 1 open reading frame 66	31
NM_016548	GOLM1	Golgi membrane protein 1	32
NM_017612	ZCCHC8	ZCCHC8	33
NM_017760	NCAPG2	Non-SMC condensin II complex, subunit G2	34
NM_020470	YIF1A	Yip1 interacting factor homolog A (*S. cerevisiae*)	35
NM_024629	MLF1IP	MLF1 interacting protein	36
NM_030819	GFOD2	Glucose-fructose oxidoreductase domain containing 2	37
U07802	ZFP36L2	Zinc finger protein 36, C3H type-like 2	38
AB014607	ACOT11	Acyl-CoA thioesterase 11	39
AK021842	ZNF362	Zinc finger proteing 362	40

in the training set. Values of E(w) started rising after 7000 iterations (over fitting). Therefore, we stopped training of the network at 7000 iterations due to optimum results obtained. The E(w) for the number of iterations descends gradually to 1500 iterations and then slowly up to 7000 iterations. The network has converged in 7000 iterations with a minimum training error of 0.22156 shown in Fig. 2

1-3. Learning Rate α and the Momentum Factor β

The learning rate α was selected between the range of 0.2 to 1 for neural network training. When the learning rate α was given 0.2, the training error obtained was 0.048. After rigorous training the error declined up to when α=0.50 and the error was at 0.0226. Then the network was stabilized with α=0.51 holding the error 0.0221. The momentum factor β was selected in the range of 0.1 to 0.9 and the training, performance was significant when β= 0.00799. Now the network was well trained with the better recalling ability. Table 2 shows the parameters chosen for the network after training.

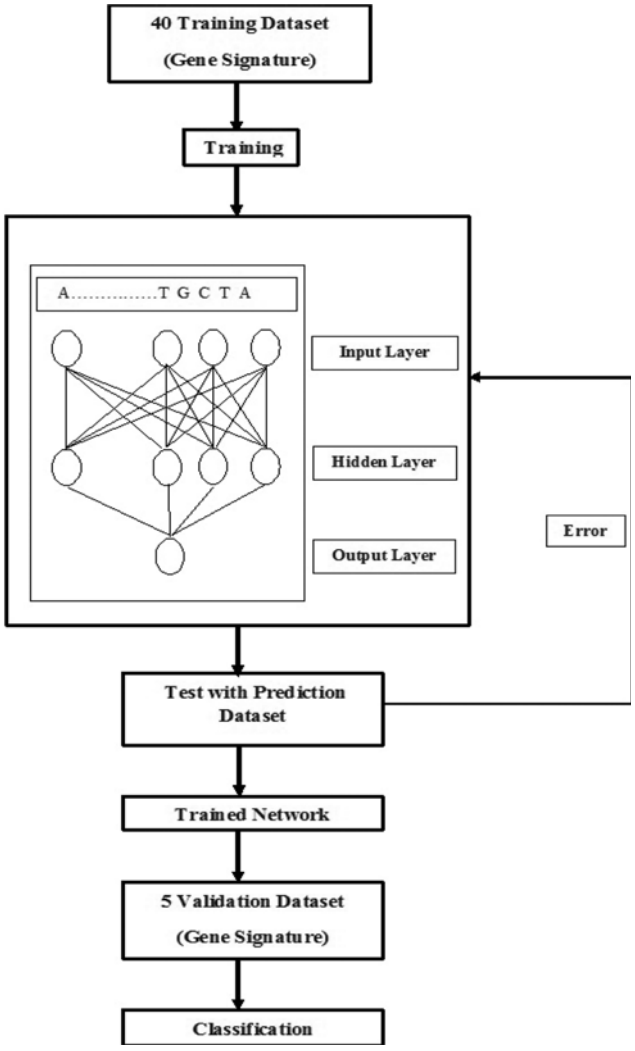

Fig. 1. Experimental architecture.

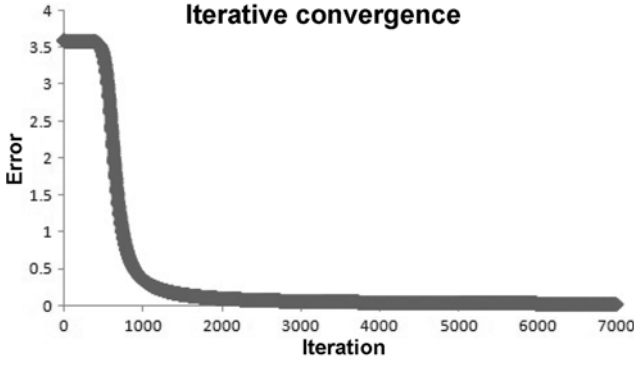

Fig. 2. Convergence criteria Iterations versus error.

1-4. Generalization Ability

The parameters for the network were selected so that they provided better learning and generalization ability. The multi input, single output neural network was configured by choosing the network parameters as α=0.51 and β=0.0079 with four hidden nodes. The training and experimental results are plotted in Fig. 3. The

Table 2. Training parameters

Training parameters	Neural network model
α	0.51
β	0.0079
Iterations	7000
Hidden nodes	4
Minimum training error	0.22156

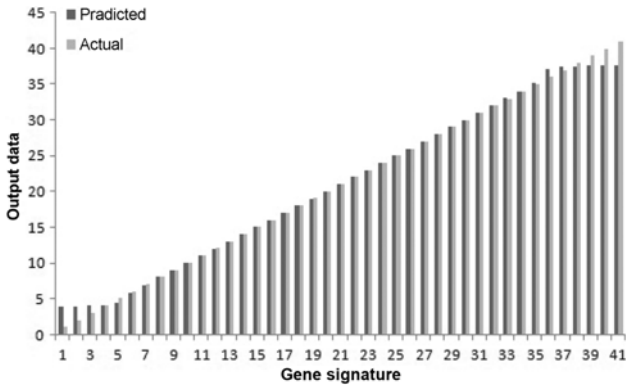

Fig. 3. Training the neural network with 40 gene sequences- actual versus predicted.

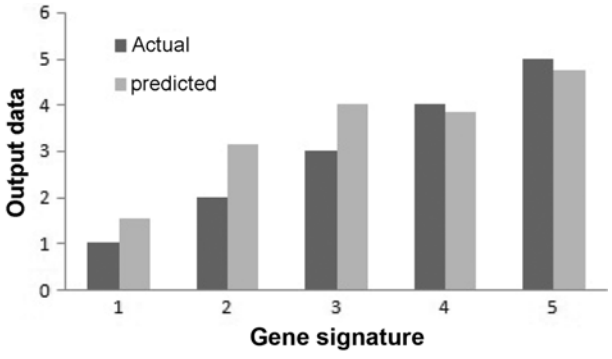

Fig. 4. Testing/prediction the five unknown sequences with the optimum weights from the trained neural network.

trained and learned networks were then subjected to assess their recall and generalization abilities. The recall ability of the trained networks is evaluated by using the same input sequences as used for training. The neural network model exhibited 100% recall ability. The generalization ability of the NN modelling, configuration is evaluated by using the performance measures, namely, mean-squared error (MSE). The MSE is defined by:

$$\mathrm{MSE}=\frac{1}{n}\sum_{k=1}^{n}(d_{Pk}-y_{Tk})^2$$

where y_{Tk} and d_{Pk} are the target and the predicted output values for the input sequences that are involved in training and n is the number of data sequences used for prediction. The MSE value for ANN is evaluated as 0.0026751.

2. Validation

The classification efficiency of ANN is tested by using the gene

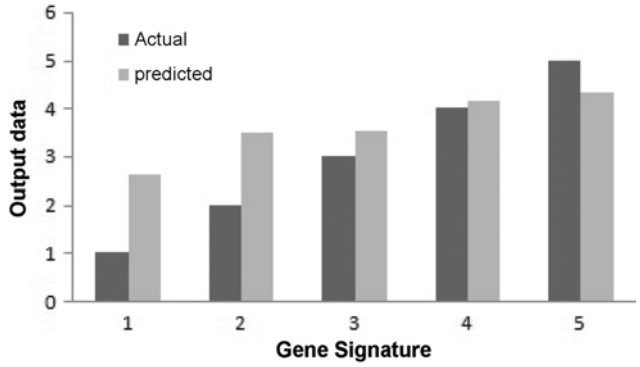

Fig. 5. Validating the five unknown sequences with the optimum weights from the trained neural network.

sequences that are not involved in training, i.e., validation set. When an untrained DNA sequence of a different breast cancer is used as input, the proposed model predicted the species corresponding to that sequence precisely. Fig. 4 shows the network prediction results against their target values for each of the untrained sample sequences. The close agreement between the predicted results and the target values in Fig. 5 indicates the better classification efficiency of the proposed neural network model.

3. Validating Models using Bootstrapping Validation

With the total of 40 gene signatures, 10 rounds of bootstrapping were performed. For each gene, MSE values were calculated. The 40 gene signature subjects from the combined dataset were randomly assigned as the training samples or the test samples for 40 rounds with 5 validation samples in each dataset. We validated the full *n*-subject model. Bootstrap can be used if either the model is completely specified, except if the parameter values or the model fitting process is entirely automated, including automatic variable selection and automatic transformation-finding procedure or a data reduction procedure. So, what we end up bootstrapping is the entire process used to fit the model as shown in Fig. 6. Actually, a good use of the bootstrap for validation purposes is in estimating the *optimism* of an index of predictive accuracy, like MSE=0.0026. Optimism is a positive bias in predictive accuracy, which usually occurs when we assess predictive accuracy on the same data set used to fit the model.

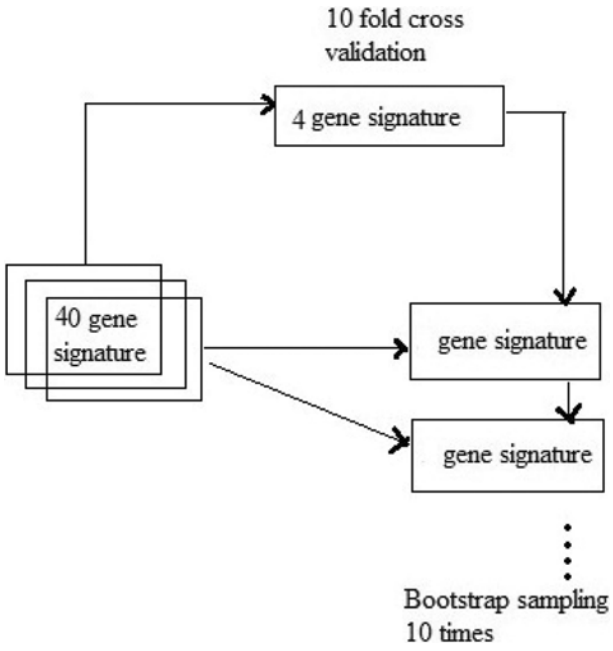

Fig. 6. Bootstrap validation.

The validity of model is the stability and reasonableness of the coefficients, the plausibility and usability of the objective function and ability to generalize inference drawn from the regression analysis. Validation is an important step in the modelling process that helps in assessing the reliability of models before they can be used in decision making. This research work, therefore, seeks to study the regression model validation process by bootstrapping approach and data splitting techniques. We reviewed regression model validation by comparing predictive index accuracy of data splitting techniques and residual resampling bootstraps. Various validation statistics such as the MSE, Mallow's cp and R^2 were used as criteria for selecting the best model and the best selection procedure for each data set. The study shows that bootstrap provides the most precise estimate of R^2 which reduces the risk of overfitting models than in data splitting techniques.

4. Overlapping Genes

Many different studies have been done on gene expression profiling for diagnosis of cancer since it is the gene expression ability which follows an undesirable track leading to mutation and hence cancer. There are certain gene markers which are differentially expressed in cancer cell as compared to normal cells. The candidate genes chosen from Wang et al. [1] study were also studied by Chou et al. [28] along with other breast cancer microarray datasets. After integrating the results of all the datasets they found ZFP36L2 and PLK1 gene from Wang et al. [1] as the prominent gene signatures. PLK1 was reported to be most relevant because it is involved in the DNA damage checkpoint response at the G2/M phase of the cell cycle and plays a role in regulating cell cycle progression. In our study we have very accurately classified these candidate genes for robust prediction of breast cancer. Hence the PLK1 gene along with other gene signatures was classified and cross validated efficiently.

CONCLUSION

The study conducted by Wang et al. [1] revealed gene signatures that can predict distant tumor recurrence, which could be applied to all lymph-node-negative patients independently of age, tumor size and grade, and ER status. Their findings explained the superior performance of this signature compared over other prognostic factors. ANN models have been shown to be a promising diagnostic tool in reducing the workload on clinicians by providing decision support system. All medical progress depends upon correct diagnosis. The present study developed, by using ANN, an efficient methodology for gene signature identification and breast cancer patients' classification as low risk or high risk patient of tumor recurrence based on the gene expression profile. This method of diagnostic classification of breast cancer from their gene-expression signatures efficiently categorizes the gene signatures and helps

in decision making for selecting appropriate therapeutic approach for the clinical management of breast cancer. The low risk patient can be recognized and unnecessary systemic adjuvant chemotherapy can be avoided. Performance of prediction result was improved with an increasing number of neurons in the hidden layer until proximity was obtained. Apparently, the prediction performance is enhanced progressively with the number of iterations until a limit is reached. Finally, we observed an efficient performing network that was able to decipher complex sequences and classify them into distinct categories, i.e., a well trained neural network with the better recalling ability providing better prediction and validation was obtained.

ACKNOWLEDGEMENT

We are thankful to National Institute of Technology Raipur for permitting us to do this work.

REFERENCES

1. Y. Wang, J. G. M. Klijn, Y. Zhang and A. M. Sieuwerts, *Lancet.*, **365**, 671 (2005).
2. Ch.Venkateswarlu, K. Kiran and J. S. Eswari, *Appl. Artif. Intell.*, **26**, 903 (2012).
3. J. S. Eswari, M. Anand and Ch. Venkateswarlu, *J. Chem. Technol. Biotechnol.*, **88**, 271 (2013).
4. J. S. Eswari and Ch. Venkateswarlu, *Int. J. Pharm.*, **4**, 465 (2012).
5. J. S. Eswari and Ch. Venkateswarlu, *Chem. Eng. Commun.*, In Press (2015).
6. J. S. Eswari and Ch. Venkateswarlu, *Environ. Eng. Sci.*, **30**, 527 (2013).
7. Y. S. Kim, S. J. Hwang, J. M. Oh, G. D. Whang and C. K. Yoo, *Korean J. Chem. Eng.*, **26**, 969 (2010).
8. N. Banerjee and J. Park, *Korean J. Chem. Eng.*, **32**, 1207 (2015).
9. Z. İlbay, S. Şahin and K. Büyükkabasakal, *Korean J. Chem. Eng.*, **31**, 1661 (2014).
10. B. Zarenezhad and A. Aminian, *Korean J. Chem. Eng.*, **28**, 1286 (2011).
11. M. Molashahi and H. Hashemipour, *Korean. J. Chem. Eng.*, **29**, 601 (2012).
12. J. Khan, J. S. Wei, M. Ringnér, L. H. Saal, M. Ladanyi and F. Westermann, *Nat. Med.*, **7**, 673 (2001).
13. Y. T. Chang, C. S. Huang, C. T. Yao and S. L. Su, *World J. Gastroenterol.*, **20**, 14463 (2014).
14. Y. T. Chang, C. T. Yao, S. L. Su and Y. C. Chou, *World J. Gastroenterol.*, **20**, 17476 (2014).
15. C. Lin, C. M. Chu, J. Lin, H. Y. Yang and S. L. Su, *PLOS One.*, **10** (2015).
16. C. M. Chu, C. J. Chen, D. C. Chan and H. S. Wu, *World J. Surg. Oncol.*, **12**, 80 (2014).
17. C. H. Lai, N. F. Chu, C. W. Chang and S. L. Wang, *PLOS One.*, **8**, 12 (2013).
18. L. J. Van't Veer, H. Dai, M. J. Van de Vijver and Y. D. He, *Nature.*, **415**, 530 (2002).
19. A. B. Dor, L. Bruhn, N. Friedman and I. Nachman, *J. Comp. Biol.*, **7**, 559 (2000).
20. S. Ramaswamy, K. N. Ross, E. S. Lander and T. R. Golub, *Nat. Genet.*, **33**, 49 (2003).
21. F. Amato, A. López, E. M. Peña-Méndez and P. Vaňhara, *J. Appl. Biomed.*, **11**, 47 (2013).
22. H.-Y. Chuang, E. Lee, Y.-T. Liu, D. Lee and T. Ideker, *Mol. Syst. Biol.*, **3**, 140 (2007).
23. L. E. Peterson, M. Ozen, H. Erdem, A. Amini, L. Gomez and C. C. Nelson, *IEEE*, **1** (2005).
24. P. J. G. Lisboa, *Neural Netw.*, **15**, 11 (2002).
25. H. T. Siegelmann and E. D. Sontag, *Appl. Math. Lett.*, **4**, 77 (1991).
26. J. L. Balcazar, *IEEE*, **7141**, 14 (1993).
27. H. L. Chou, C. T. Yao, S. L. Su and C. Y. Lee, *BMC Bioinfo.*, **14**, 100 (2013).