Classification of Acute Upper Gastrointestinal Bleeding based on Rough Neural Network

Su Mon Aung, Win Pa Pa
University of Computer Studies, Yangon
sumonaungucsy@gmail.com,winpapa76@gmail.com

Abstract

The main emphasize of paper is the classification of Acute upper Gastrointestinal Bleeding based on Rough neural network. Classification is used to extract model describing important and data classes or future data trends. A conventional neural network consists of several layers of neurons. Each neuron receives input from other neurons and external environment and produces output. A rough neural network consists of conventional neurons and rough neurons connected to each other. A rough neuron can be viewed as a pair of neurons, one for the upper bound and the other for the lower bound. Rough neural network consists of one input layer, one output layer and one hidden layer. The system can classify 7 types of classes for acute upper gastrointestinal bleeding. This system is implemented by using Java programming language.

Keywords
Neural network, neurons, rough neural network, Acute upper gastrointestinal bleeding

1. Introduction

Classification is the process of finding a model (or function) that describes and distinguishes data classes or concepts, for the purpose of beings able to use the model to predict the class of objects whose class label is unknown. The derive model may be represented in various forms, such as classification rules, decisions trees, mathematical formulae, or neural network. Data mining refers to extracting or mining knowledge from large amounts of data. In many applications of medical data mining [4], the interest focuses on the effectiveness of certain treatments. The previous history of patients and their treatment must be considered. It may require a medical specialist to understand medical data. The concept of neural networks [2] started in the late 1800s as an effort to describe how the human mind performed. The Austrian School of economics theory of spontaneous order was explained by Murray Rothbard to have been first realised by Zhuangzi (Chuang Tzu) who said, "Good order results spontaneously when things are let alone."[3] In the brain spontaneous order arises out of decentralized networks of simple units (neurons). In the late 1940s Donald Hebb made one of the first hypotheses of learning with a mechanism of neural plasticity called Hebbian learning. Hebbian learning is considered to be a 'typical' unsupervised learning rule and it and later variants were early models for long term potentiation. These ideas started being applied to computational models in 1948 with Turing's B-type machines and the perceptron.

The perceptron is essentially a linear classifier for classifying data and an output function f = w' x + b. Its parameters are adapted with an ad-hoc rule similar to stochastic determining the optimal parameters in a model of this type is not trivial, and steepest gradient descent methods cannot be relied upon to give the solution without a good starting point. In recent times, networks with the same architecture as the backpropagation network are referred to as Multi-Layer Perceptrons. This name does not impose any limitations on the type of algorithm used for learning.

The backpropagation network [1] generated much enthusiasm at the time and there was much controversy about whether such learning could be implemented in the brain or not, partly because a mechanism for reverse signaling was not obvious at the time, but most importantly because there was no plausible source for the 'teaching' or 'target' signal.

Artificial Neural Network (ANN) is an interconnected group of artificial neurons that uses a mathematical or computational model for information processing based on a connectionist approach to computation. The networks were inspired by the structure and operation of biological neurons. Knowledge is stored in the topology of the network itself rather than in explicitly coded data structures. Neural networks are composed of many simple processing units or artificial neurons joined through numerous interconnections. These neurons are usually organized into groups called layers. The input layer is connected to the output layer through junctions with a hidden layer. The network learns by a process involving the modification of the connection weights between neurons and layers. ANN can be classified as either feed forward, recurrent, modular, stochastic and many others, depending on how data is processed through the network. The feed forward neural networks are the first and simplest type of neural networks.

ANN is used to train input data so that it can generate the appropriate output according to the desired target. Before the training process starts, all weights must be initialized to small random numbers. This is to make sure that the network is not saturated by large values of the weights. The steps are as follows:(1) Select the training pair from training set, applying the input vector to the network input.(2) Calculate the output of the network.(3) Calculate the error between the network output and the desired output.(4) Adjust the weights of the network in a way that minimizes the error.(5) Repeat step 1 through 4 until the error is acceptably low.

The concept of upper and lower bound [6] has been used in a variety of applications in artificial intelligence. In particular, theory of rough sets has demonstrated the usefulness of upper and lower bounds in rule generation. Each value in a rough pattern is a pair of upper and lower bound. The conventional neural network models need to be modified to accommodate rough patterns.

In this paper, we classified the acute upper gastrointestinal bleeding based on rough neural network. This is the most common gastrointestinal emergency. In Myanmar, nearly about 0.21 percent of hospital admission is acute upper gastrointestinal bleeding. The main symptoms of upper gastrointestinal bleeding are haematemesis and melaena. Haematemesis is effortless vomiting of coffee ground material or bright red blood (not associated with cough, forthy and respiratory symptom). Melaena is passing of black tarry stool per rectum, sticky offensive smell, change to red colored when wash with water.

The remainder of this paper is organized as follow as Section 2 is Related Work. Section 3 is details described rough neural network and backpropagation algorithm. Section 4 is implementation of the system and experimental result of training data and testing data. Section 5 is provide conclusion remark.

2. Related Work

Neural Network based Medical Diagnosis Systems: Medical Diagnosis using Artificial Neural Networks (ANN)[7] is currently a very active research area in medicine and it is believed that it will be more widely used in biomedical systems in the next few years. This is primarily because the solution is not restricted to linear form. Neural Networks are ideal in recognizing diseases using scans since there is no need to provide a specific algorithm on how to identify the disease. Neural networks learn by example so the details of how to recognize the disease are not needed. The quantity of examples is not as important as the 'quality'. The examples need to be selected very carefully if the system is to perform consistently and efficiently. The eight input nodes represent features of classifications, areas in breast tissue where tiny calcium deposits build up and might indicate the presence of cancer. The AI system feeds these nodes into the neural network to provide a statistical indication of the possibility that a group of calcifications is malignant.

Rule Based Medical Diagnosis Systems: MYCIN was an expert system developed to diagnose blood infections. MYCIN used about 450 rules and was able to perform as well as some experts and considerably better than junior doctors. No general theoretical model existed from which rules were deduced. They had to be acquired from extensive interviewing of experts who in turn acquired them from direct experience of cases. The rules also had to reflect the uncertainty associated with medical knowledge. To this end, MYCIN incorporated a calculus of certainty factors which seemed to fit well with how doctors assess the impact of evidence on the diagnosis.

Giger and Huo (1999) used artificial neural network (ANN) [9] to develop a CAD that incorporates various computer-extracted image features from mammogram images to differentiate malignant from benign masses. The performance of the computer aid with 100% sensitivity was appreciable with a positive predictive value of 83%, which was 12% higher than an experienced mammographer. Wu et al. (1993)[10] used Multi-layered Perceptron (MLP) network to differentiate between malignant and benign mammographic patterns based on radiographic features extracted by radiologists. The ability of neural network classification was then compared to classification by mammographers and the results showed that the capability of the neural network to do classification is better than that of a mammographer alone.

Other researches in neural networks implementation in cancer diagnosis have been done by Kok et. al. (1999)[8], Mitra et. al. (2000) and Mashor et. al (2004) for cervical cancer and Yao & Liu (1999) and Kates et. al.(2000) for breast cancer. Yao & Liu (1999) defined two neural network approaches for breast cancer diagnosis, evolutionary and ensemble. The evolutionary approach was used to design compact neural networks automatically by evolving network architectures and weights, while the ensemble approach was aimed at tackling large problems that may not dealt with efficiently by a monolithic neural network. Kates et. al (2000) presented the potential contributions of neural network to a clinical decision support framework for the prediction of breast cancer therapy response.

Real life applications: [4] the tasks to which artificial neural networks are applied tend to fall within the following broad categories:

Function approximation, or regression analysis, including time series prediction and modeling. Classification, includes pattern and sequence recognition, novelty detection and sequential decision making. Data processing, includes filtering, clustering, blind signal separation and compression.

Application areas of ANNs include system identification and control (vehicle control, process control), game-playing and decision making (backgammon, chess, racing), pattern recognition (radar systems, face identification, object recognition, etc.), sequence recognition (gesture, speech, handwritten text recognition), medical diagnosis, financial applications, data mining, visualization and e-mail spam filtering.

3. Rough Neural Network

A rough neural network [6] consists of conventional neurons and rough neurons connected to each other. A rough neuron r in rough neural networks can be viewed as a pair of neurons, one for the upper bound and the other for the lower bound. Rough neural networks consist of one input layer, one output layer and one or more hidden layer.

This system uses multi-layered, feed-forward, and backpropagation design outlined to describe the methodology of rough neural networks. Rough neural network used in this paper consists of one input layer, one output layer and one hidden layer of rough neurons.

In Figure 1, the input layer neurons accept input from the external environment. The output from input layer neurons is feed to the hidden layer neurons. The hidden layer neurons feed their output to the output layer neurons which send their output to external environment. The training and testing stage in the development of a rough neural network is similar to the conventional neural network. In this paper, we used rough neural network because rough neural network is faster than the conventional neural network.

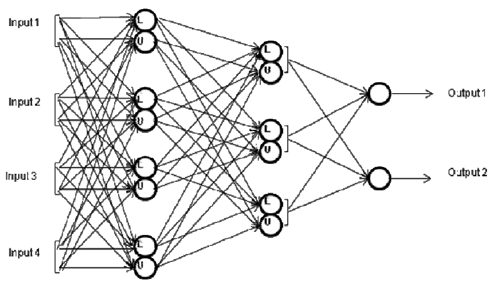

Figure1. Rough Neural Network

3.1. Rough Neural Network Equations

The input of a conventional, lower or upper neuron is calculated using the weighted sum is:

$$\mathbf{input}_i = \sum \mathbf{w}_{ij} \times \mathbf{output}_j \quad (3.1)$$

where i and j are either the conventional neurons or upper, lower neurons of a rough neuron.

The sigmoid function [6] is a commonly used activation function. The output from Sigmoid function falls in a range from 0 to 1.

$$\mathbf{transfer(u)=1/(1+e^{-u})} \quad (3.2)$$

The output of a rough neuron is calculated using a transfer function as:

$$\mathbf{output}_U = \mathbf{max(transfer(input}_U\mathbf{),transfer(input}_L\mathbf{))} \quad (3.3)$$

$$\mathbf{output}_L = \mathbf{min(transfer(input}_U\mathbf{),transfer(input}_L\mathbf{))} \quad (3.4)$$

$$\mathbf{output}_i = \mathbf{(output}_U - \mathbf{output}_L\mathbf{)/average(output}_U\mathbf{,output}_L\mathbf{)} \quad (3.5)$$

The output of a conventional neuron i is simply calculated as :

$$\mathbf{output}_i \mathbf{=transfer\ (input}_i\mathbf{)} \quad (3.6)$$

Calculate output layer error : $\alpha_j = (t_j - o_j)$ **(3.7)**

Calculate hidden layer error : $\alpha_k = \sum \alpha_k w_k$ **(3.8)**

Adjust weight values according to error function until the minimal error is achieved.

Network includes inputs and target outputs. Initially we needed to do three preprocess. They are preprocess data scaling, initialize target value and initialize weights value for upper and lower bound of rough neurons.

Preprocess data

To scale the inputs and targets data to ensure they always fall within a specified range.

Initialize the weight values

To avoid the result located in flat area, random weight values are selected.

Initialize target value

The learning process used in this paper is called supervised learning. In supervised learning, the desired output is known for output layer neurons for the examples in the training set. The network attempts to adjust weights of the connections between neurons to produce the desired output.

In the testing stage, the network is tested for another set of examples for which the output from the output layer neurons is known.

3.2. Backpropagation Algorithm

The backpropagation algorithm [1] defines two sweeps of the network: first a forward sweep from the input layer to the output layer, and then a backward sweep from the output layer to the input layer.

The forward sweep propagates input vectors through the network to provide outputs at the output layer. The backward sweep is similar to the forward sweep except that error values are propagated back through the network to determine how the weights are to be changed during training. During the backward sweep, values pass along the weighted connections in the reverse direction to that which was taken during the forward sweep.

Backpropagation algorithm is as follows:

Step 1: Read input pattern and associated output pattern. SET STOP=TRUE

Step 2: For input layer- assign as net input to each unit its corresponding element in the input vector. The output for each unit is its net input.

Step 3: For the first hidden layer unit – calculate the net input and output. Repeat step 3 for all subsequent hidden layers.

Step 4: For the output layer unit – calculate the net input and output.

Step 5: Is the difference between target and output pattern within tolerance?
If NO then STOP=FALSE

Step 6: For each output unit calculate its error:

Step 7: For last hidden layer calculate error for each unit. Repeat step 7 for all subsequent hidden layers.

Step 8: For all layers update weights for each unit.

4. Implementation of the System

In this paper, classify acute upper gastrointestinal bleeding based on rough neural network to get with minimum error. The backpropagation algorithm is used for classifying.

The system can classify 7 types of classes. They are

(1) Bleeding peptic ulcer
(2) Carcinoma stomach
(3) Carcinoma oesophagus
(4) Bleeding oesophageal varies
(5) Gastric erosion
(6) DHF (Dengue haemorrhagic fever) and
(7) Haematological disorder

The system has five main components. They are

(1) input symptoms
(2) prescaling inputs
(3) weights
(4) Target value
(5) Rough neural network

A set of input symptoms are sent from user to the system. Prescaling input component which scaled input symptoms by using equations. A rough neuron in rough neural network has a pair of neurons, one for upper and other for lower. And we want positive and negative weights. Weights are initialized by using uniform random methods. We used the range value of upper weights are (-0.2 to 0.8) and range value of lower weights are (-0.8 to 0.2). And then we set target value for categorization. Target representation is more efficient learning.

Prescaling inputs, weights and target value are given to Rough neural network. Rough neural network process forward propagation and calculate errors and backward propagation. Adjust weight values according to error function until the minimal error is achieved. Rough neural network stop the processing when the error is less than tolerance and display the results to the user. The testing data can save in the database.

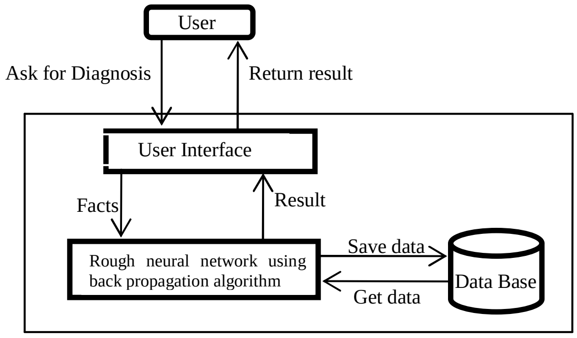

Figure2. System architecture

In Figure 3, display the forward process of rough neural network. The inputs given from user are prescaling and multiply weights of neurons. The result is given to Sigmoid function to produce output of input layer. And then the sigmoid result is chosen minimum value for lower hidden layer and maximum value for upper hidden layers as inputs. The input and hidden layer weights are multiplied and the result is given to Sigmoid function. The average value of maximum and minimum is given to the output layer as input. The input and output weights are multiplied and the Sigmoid are produce as output. This process is called forward process.

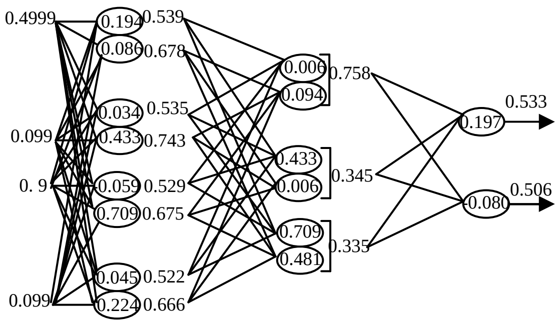

Figure3. Forward process example

For example, use four input symptoms which are 2,1,3,1. These inputs are scaled by using formula. The results are 0.4999, 0.0999, 0.9, 0.0999. And then initialize weights for lower input neurons which are 0.194, 0.034, -0.059, 0.045. Initialize weights for upper input neurons which are 0.086, 0.433, 0.709, 0.224. Weights for lower hidden layers are 0.006, 0.433, 0.709. Weights for upper hidden layers are -0.094, 0.006, 0.481. Weights for output layer are 0.197,-0.080.

Calculate the sigmoid input for input layer based on prescaling inputs and initialized weights of lower and upper neurons.A1 for input of lower input neurons and B1 for input of upper input neurons.

A1= (0.4999*0.194) + (0.0999*0.034) + (0.9* - 0.059) + (0.0999*0.045)
= 0.1577981

Sigmoid(A1)= $1/(1+e^{-A1})$ = 0.539

B1= (0.4999*0.086) + (0.0999*0.433) + (0.9* 0.709) + (0.0999*0.224)
= 0.7466483

Sigmoid(B1)= $1/(1+e^{-B1})$ = 0.678

Other input of lower input layers and upper input layers are the same process A1 and B1. They are A2, A3, A4 for lower input layers and B2, B3, B4 for upper input layers. A2, A3 and A4 are 0.535, 0.529 and 0.522 respectively. B2, B3 and B4 are 0.743,0.675 and 0.666 respectively.

And then input of upper hidden neurons and input of lower hidden neurons are chosen. y1 for minimum of sigmoid function and y2 for maximum of sigmoid function of input layers.

y1(min)=min(A1,B1)= min(0.539,0.678)=0.539
y1(max)=max(A1,B1)= max(0.539,0.678)=0.678

Calculate the sigmoid input for hidden layer based on output of input layers and initialized weights of hidden layers. C1 is input of lower hidden neurons and D1 is input of upper hidden neurons.

C1= (0.539*0.006) + (0.535*-0.073) + (0.529* 0.200) + (0.522*0.195)
= 0.171769

Sigmoid(C1)= $1/(1+e^{-C1})$ = 0.543

D1= (0.678*-0.094) + (0.743*0.006) + (0.675* 0.481) + (0.666*0.445)
= 0.561771

Sigmoid(D1)= $1/(1+e^{-D1})$ = 0.636

Other input of lower hidden layers and upper hidden layers are the same process C1 and D1. They are C2 and C3 for lower hidden layers and D2, D3 for upper hidden layers. C2, C3 are 0.543, 0.527 respectively. D2 and D3 are 0.769 and 0.739.

And then calculate input of output layer neurons. H1 is output of sigmoid function.

H1=(D1-C1) / avg(C1,D1)
= (0.636-0.543) / (0.5895) = 0.758

Other outputs of sigmoid function are H2 and H3. The value of H2 and H3 are 0.345 and 0.335.

The sigmoid input for output layer is also calculate based on output of hidden layers and initialized weights of output layers. O1 for input of output neurons .

O1=(0.758*0.197)+(0.345*-0.080)+(0.335*0.388)
= 0.133506

Sigmoid(O1)= $1/(1+e^{-O1})$ = 0.533

O2=0.506 for output layer.

In Figure 4 display the backward process of rough neural network. Error of output layer are calculated and delta value to change the value of weights. The error of output layer and weights of output layer are multiplied to find the error of hidden. The same process is done until the input layer. And then change the weights of layers based on delta values. This process is called backward processing. The process is stopped when the error of output layer is less than tolerance.

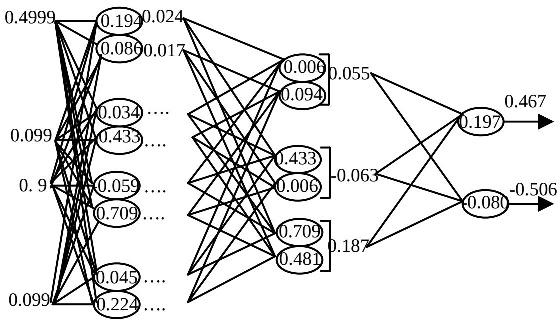

Figure4. Backward process example

Above example, calculate the forward processing. Now calculate the error for backward processing. In this paper, the supervised learning are used. The target value for above example is 1. Calculate the error of output layers. E8 and E9 are errors of output layer.

E8 = T- O1 = 1- 0.533 = 0.467
E9 = T- O2 = 0- 0.506 = -0.506

Calculate hidden layers error. They are E5, E6 and E7.

E5 = (E8* 0.197) + (E9*0.073) = 0.055

The values for E6 and E7 are -0.063 and 0.187 respectively. Also calculate input layer errors. They are E1,E2,E3 and E4.

E1(l) = (E5*0.006) + (E6*0.083) + (E7*-0.063)
= -0.017

E1(u) = (E5*-0.094) + (E6*0.799) + (E7*0.427)
= 0.024

E2, E3 and E4 are the same process as E1.

And then change the old weight of input layer based on the delta value and learning rate.Using the learning rate value for this system is 0.1.

Delta value = E1(l) * x1* (1-x1)
= -0.017*0.499*(1-0.499)
= -0.0042

The updated weight of input layer is W1(l) and W1(u).

W1(l) = W1L(old) + learning rate * Delta value
= 0.194 + 0.1 * -0.0042
= 0.193

Delta value = E1(u) * x1* (1-x1)
= 0.024*0.499*(1-0.499)
= 0.0059

W1(u) = W1U(old) + learning rate * Delta value
= 0.086 + 0.1 * 0.0059
= 0.0864

The backward process is continued until the error is less than tolerance. In this paper, Using tolerance value for this system is 0.1. The input symptoms attributes are listed in Table 1.

Table1. Description of attributes

field	field
ID	excessive salivation
pain in epigestrium	fever
right hypochondrium pain	fever nature
pain	weight
nature of pain	anaemia
after meal	red spots
mass in epigestrium	neck glands
food habit	constitutional
habit of	shock
taking NSAIDS	hyperbilirubinaemia
acute massive	easy to bleed
stress	organomegaly
appetite	oedema
progress dyshpagia	age

4.1. Experimental Results

In the experiment, acute upper gastrointestinal bleeding is classify based on rough neural network. To classify seven types of acute upper GI bleeding, 27 neurons for input layer, 6 neurons for hidden layers and 7 neurons for output layer are used in this system.

Three hundred training data are used to classify for acute upper gastrointestinal bleeding. In the processing, correct training data and incorrect training data are counted to calculate accuracy. In this experiment, accuracy is 0.87 or 87 percent for training data. And one hundred and five testing data are used for system to classify. The accuracy of testing data is 0.862 or 86.2 percent.

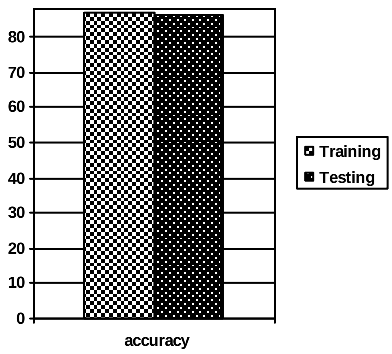

Figure5. Compare accuracy of training data and testing data

In Figure 5, the accuracy of training and testing data are compared. The system can classify 86 percent for acute upper gastrointestinal bleeding.

5. Conclusion

This paper emphasizes to classify the Acute Upper Gastrointestinal Bleeding by applying rough neural network. When using rough neural network, the classification itself can be easily validated. Rough neural network use a combination of rough and conventional neurons. A rough neuron can be viewed as a pair of neurons. One neuron corresponds to the upper bound and the other corresponds to the lower bound. The error in estimation from rough neural network model is significantly lower than the conventional neural network model.

6. References

[1] Ackly., D.H., Hinton, G.E and Sejnowski, T.J.(1985) A learning algorithm for Boltzmann machines. Cognitive Science, 9:1s47-169

[2] Adamo and P.J,Lingras,M.(1995). Estimation of AADT Volume Using Neural Networks, Computing in Civil and Building Engineering, Pahl & Werner (Eds.), 1355-1362

[3] Baum, E.B. and Haussler, D. What size net gives valid generalization?Neural Computation, (1989), 1:151-160

[4]Canada,DeGarmo, E.P., Sullivan, W.G,J.R. Engineering Economy, Macmillan Publishing Co., New York, N.Y., 1984, pp.264-266

[5] Dorffner, G.(ed.)(1997) Neural Networks and a New Artificial Intelligence. Landon: International Thomson Computer press

[6] Garber, N.J. and Hoel, L.A.1988. Traffic and Highway Engineering, West Publishing Co., New York, N.Y.,pp.97-118

[7] J. B. Siddharth Jonathan and K.N. Shruthi, Department of Computer Science, Stanford University, USA and Department of Biomedical Engineering, University of Southern California, USA.

[8]Kok M.R.,Schreiner-KokP.G.,VeenV.D.,and BoonM.E.,(1999),"Potentially difficult smears of womenwith squamous cell carcinoma pose fewer problems when PAPNET is used for primary screening",Cytopathology, 10(5),pp 324-34.

[9] M.Y. Mashor, N.A. Mat Isa and N.H. Othman, Electronic & Biomedical Intelligent Systems (EBItS) Research Group, School of Mechatronic Engineering ,Kolej Universiti Kejuruteraan Utara Malaysia, 02600 Jejawi, Arau, Perlis, MALAYSIA

[10]Wu,Y.,Giger,M.L.,Doi,K.,Vyborny,C.J.,Schmidt,R.A., Metz,C.E.(1993).ANNs in mammography:Application to decision making in the diagnosis of breast cancer,Radiology.Pp.81-87.