

Current Developments in the Analysis of Proteomic Data: Artificial Neural Network Data Mining Techniques for the Identification of Proteomic Biomarkers Related to Breast Cancer

Lee J. Lancashire¹, Shahid Mian¹, Ian O. Ellis², Robert C. Rees¹ and Graham R. Ball^{1,*}

¹The Nottingham Trent University, School of Science, Clifton Campus, Clifton Lane, Nottingham. NG11 8NS. UK and

²Department of Histopathology, Nottingham City Hospital, Hucknall Road, Nottingham. NG5 1PB. UK

Abstract: Artificial Neural Network (ANN) techniques are becoming increasingly popular in many areas of the biological sciences for the analysis of complex data. Careful selection of key parameters when developing ANN models and algorithms is extremely important in order to create generalised models with real-world applicability. This study applies these approaches to the analysis of proteomic data generated using Surface Enhanced Laser Desorption/Ionisation mass spectrometry profiling of cell lines from patients with breast cancer. Examples of these approaches include constrained architecture, Correlated Activity Pruning (CAPing), appropriate training termination methods and other, more advanced methodologies such as parameterisation by weightings analysis and stepwise additive approaches. These approaches, when applied to breast cancer cell lines from actual patients, resulted in the identification of 8 protein/peptide molecular ions which were capable of classifying samples into their respective groups to an accuracy of 94.8 % with an area under the curve value of 0.993 when examined with a receiver operating characteristic curve. Several ions which appear to show a significant up or down-regulation with regards to treatment regimen have also been identified. These results indicate that when coupled with other powerful techniques, the development of these novel methodologies and algorithms using ANNs allows for the development of effective data mining tools in order to analyse complex, non-linear, noisy data.

Key Words: Artificial neural networks, breast cancer, methodologies, models and algorithms, data-mining, proteomics.

1. INTRODUCTION

This paper will consider current methodologies for the analysis of proteomic data using Artificial Neural Network (ANN) based methodologies, their advantages, disadvantages and limitations, and then will describe an application of novel methodologies developed using actual patient data.

ANN techniques have been widely applied to many areas of the physical sciences for the analysis of complex systems. As such, extensive knowledge exists on the application and limitations of these methods. Similarly, methodologies exist to overcome many of these limitations and enhance the predictive capabilities and real-world applicability of developed models. This study applies these approaches to the analysis of proteomic data generated using Surface Enhanced Laser Desorption/Ionisation (SELDI) mass spectrometry (MS) profiling with the aim of identifying candidate biomarkers indicative of treatment regimen for chemosensitive (MCF-7 and T47-D) breast cancer cell lines, in order to develop ANN algorithms to correctly assign samples into their appropriate class of either control or drug treated. Examples of these approaches and important parameters which need to be considered when developing ANN models will be discussed, followed by methodologies employed in order to create generalised models with real-world applicability.

1.1. SELDI-MS

Ciphergen SELDI-MS Protein Chip Technology allows for the non-destructive analysis of both large and small biomolecules. This system consists of aluminium chips which allow the selective binding of protein samples on the surface chemistry of the chip. Bound proteins are then co-crystallised with a large molar excess of matrices for Matrix Assisted Laser Desorption/Ionization Mass Spectrometry, which are small energy-absorbing molecules (usually organic acids). The chip is then placed in a SELDI instrument and the protein-matrix mixture is desorbed and ionized by pulses of UV laser. The masses of the proteins are determined from the time-of-flight of the protein molecular ions in the flight tube of the instrument (Boyle *et al.*, 2001).

This allows for the rapid, high-throughput analysis of samples containing vast amounts of proteins with excellent reproducibility. It can be used in generating patterns that these masses of proteins produce, and therefore is useful in showing the differences between patterns in which minimal protein changes may have very important significances (Zhukov *et al.*, 2003). An example of this is differential expression of proteins in diverse tissues, such as tissues during various stages of a disease (Pawletz *et al.*, 2001). Due to the vast amount of data generated by SELDI-MS, and its consequent high dimensionality (approximately 13,000 data points per sample in the 3-30 kDa mass range), robust computer algorithms are vital to screen for potential biomarkers and develop parsimonious systems (Ball *et al.*, 2002).

*Address correspondence to this author at The Nottingham Trent University, School of Science, Clifton Campus, Clifton Lane, Nottingham. NG11 8NS. UK; Tel: 44-115 848 3394; Fax: 44 115 848 3093; E-mail: graham.balls@ntu.ac.uk

1.2. Artificial Neural Networks

Artificial Neural Networks (ANNs) are a form of Artificial Intelligence (AI) capable of modelling for complex systems and are fast emerging as one of the most popular areas of current research. ANNs are loosely based on the way a biological neuron is believed to organise and process information, and therefore have many advantages. Firstly, they are essentially non-linear so that they are able to process data containing complex interactions that are usually difficult to interpret. In addition to this, they are capable of generalisation, so they can interpret information which is different to that of the training data, thus representing a “real-world” solution to a given problem. Another advantage is that they are fault tolerant, *i.e.* they have the ability of handling noisy or fuzzy information, while also being able to tolerate data which are incomplete or contain missing values. The one major limitation of ANNs is that they do not explain how they reach a conclusion, and because of this, they have often been referred to as “black boxes”. There are many different learning algorithms applied to ANNs, and many different network architectures. Some of the most popular will be discussed now, leading to a more in-depth explanation of the multi-layer perceptron (MLP) and the back-propagation (BP) algorithm. This type of ANN was used in this study because of its flexibility, adaptability and wide-application capabilities. This MLP ANN has also been shown to successfully model on this type of data in previous studies (Lancashire *et al.*, 2003; Mian *et al.*, 2003; Ball *et al.*, 2002).

Adaptive Resonance Theory (ART) neural networks, first described by Carpenter and Grossberg (1988), perform unsupervised learning. The structure of the ART model consists of an input processing field, a clustering field, and a reset subsystem. There are two sets of connections (each with its own weights) between each node in each layer. The connection weights between the layers can be modified according to different learning rules. The node in the clustering layer with the largest net input becomes the candidate to learn the input pattern. Whether this candidate will learn the input pattern is dependant upon the reset mechanism, which controls the degree of similarity between the patterns placed in the same node (Cao and Wu, 2002). During this learning method, learning is fast and is guaranteed to converge in 3 passes of any set of patterns (Palmer-Brown *et al.*, 2003).

Kohonen networks (Kohonen, 1989), also known as self-organizing feature maps, consist of just two layers, an input layer and an output layer. The output layer of these networks may be two-dimensional (Ultsch and Roske, 2002), so that these networks are used to map a three-dimensional surface onto a two-dimensional map (Barlow, 1995). The training patterns are presented to the input layer, then propagated to the output layer and evaluated, with one output neuron being labelled as the “winner”. The network weights are adjusted during training and this process is repeated for all patterns for a pre-determined number of epochs, forming clusters within the data (Ward Systems Group, Inc., 1993). These networks are unique in that they autonomously self-organize themselves and converge into a stable structure representing the information that has been learnt (Nour and Madey, 1996).

By combining Kohonen learning and Grossberg learning, a new type of mapping network was developed. This counterpropagation network functions as a statistically optimal self-adapting look-up table (Hecht-Nielsen, 1988). This integration makes it resemble the biological neural network more closely and able to solve more complex problems. The array of input neurons operates simply as a flow-through layer for the input vectors. The Kohonen layer of neurons are used to establish the clusters with samples belonging to the same category, *i.e.* there are no samples of different classes. Based on the clusters obtained by Kohonen training, the output layer performs a further training to yield an output, similar to the pre-defined target vector (Li *et al.*, 1999).

Genetic algorithms (GA) are search algorithms founded upon the principles of natural evolution proposed by Darwin (Lucasius and Kateman, 1993). A GA is an iterative procedure, and during each iteration, a set (known as a population) of individuals, each with a potential solution to the problem is maintained. The fitness of each individual is measured according to an evaluation function, then a new population is formed by selecting the fitter individuals (Kim *et al.*, 1998). The larger the breeding pool size, the greater the potential of it producing a better individual, with this process being repeated until pre-set termination criteria are met.

Radial Basis Function (RBF) neural networks were first introduced by Moody and Darkin (1989). RBF networks are linear in their parameters, therefore once suitable basis function parameters have been chosen, they can be trained using a fast linear supervised training scheme. The most common nonlinear function used in RBF networks is the Gaussian function (Cowper *et al.*, 2002), and training in RBF networks is essentially a two stage process. In the first stage, the parameters governing the basis functions (hidden units) are determined using unsupervised methods. The second stage of training involves the determination of the weights of the final layer, which requires the solution of a linear problem, and is therefore a fast process (Bishop, 1995).

The Hopfield neural network model was first described by Hopfield (1982), and like all neural networks, it comprises a set of nodes and their connections. Hopfield networks have been described as “general content-addressable memories”, because they can be trained to recall a unique pre-determined state when they are presented with information associated with that state. This enables them to be trained to distinguish between two closely related states (Pritchard and Dufton, 2000). Hopfield networks, in their original form, suffer from being a gradient descent technique incapable of escaping local minima, and are further limited by the fact that their penalty parameter approach for solving optimisation problems often results in poor quality solutions (Smith *et al.*, 2003).

A recurrent neural network (RNN) is a neural network with feedback connections. From training examples, RNNs can learn to map input sequences to output sequences. In principle, they can implement almost arbitrary sequential behaviour. A recurrent network may respond to the same input pattern differently at different times, depending upon

the patterns that have been presented as inputs just previously. Thus, the sequence of the patterns is as important as the input pattern itself. The patterns must always be presented in the same order, therefore random selection is not allowed. An extra slab is present in the input layer that is connected to the hidden layer just like the other input slab. This extra slab holds the contents of one of the layers as it existed when the previous pattern was trained. In this way the network sees previous knowledge it had about previous inputs. This extra slab is sometimes called the network's "long-term" memory (Ward Systems Group, Inc., 1993).

1.3. Learning

Learning in biological organisms involves the adjustments of synaptic connections which exist between the neurons, and the same happens in ANNs. They are capable of learning either supervised (where the inputs, with known outputs are used for training), for example with the BP algorithm, or unsupervised (where only input information is used for training), as in Kohonen's self organizing maps (Niculescu, 2003). For the purpose of this manuscript, the BP algorithm will be described. During supervised learning in ANNs, the network architecture is modified, adjusting the weights of the links between the layers as the network is presented with data examples so as to learn the mathematical relationships between an input and its associated output. A learning rule defines how the network weights should be adjusted between each training cycle (epoch). In supervised learning, an error-correction learning rule is used which calculates the difference in error between the ANN output value and that of the correct known output value and uses this to modify the connection weights as to iteratively reduce the overall network error.

1.4. The Multi-Layer Perceptron for Data Mining of SELDI-MS Data

The basic unit of an ANN is known as a node and these nodes can form several layers that are arranged so that each node in one layer is connected with each node in the next

layer, with the entire group of connected nodes constituting the complete ANN, often referred to as the MLP, as shown in Fig. (1). The global architecture of a MLP network is made up of nodes arranged into one input layer, one output layer and at least one hidden layer. The input layer of the network receives input from the original data corresponding to one of the variables used as an input, the hidden layer serves as a feature detector, while the output layer outputs its data as the results. The purpose of the hidden layer is to allow the ANN to be able to cope with non-linear data and thus giving it the ability to solve non-linear classification problems.

ANNs learn through special training algorithms such as the BP algorithm. It is well documented that MLPs are commonly used in conjunction with the BP algorithm for the data mining of complex data (Wei *et al.*, 1998; Balls *et al.*, 1996; Desilva *et al.*, 1994). The reason for this being that it is able to cope with data containing high levels of background noise and can be utilised in both identifying specific markers which may be responsible for the classification of certain disease states, and also in identifying the influence of interacting factors between these markers. The task of classifying data is to decide class membership of an unknown data item based on a data set of items with known class membership (Dreiseitl and Ohno-Machado, 2002). The BP training algorithm consists of two main steps, a forward step which involves generating a solution to the classification problem in question, and then a back propagation of the error to modify the weights in the direction of minimum error. The forward step presents the inputs as training examples to the ANN, which is passed to each hidden node in the hidden layer where the activation at that node is calculated using a sigmoidal transfer function (Rumelhart and McClelland, 1986) in order to yield an output which is transferred to the output layer. The activations calculated at the output signify the ANNs solution to the problem. In the backward step, the difference in error between the ANN output and the target output is calculated and used to adjust the values of the connecting weights, from the output, through the hidden layer, to the input layer (Basheer and

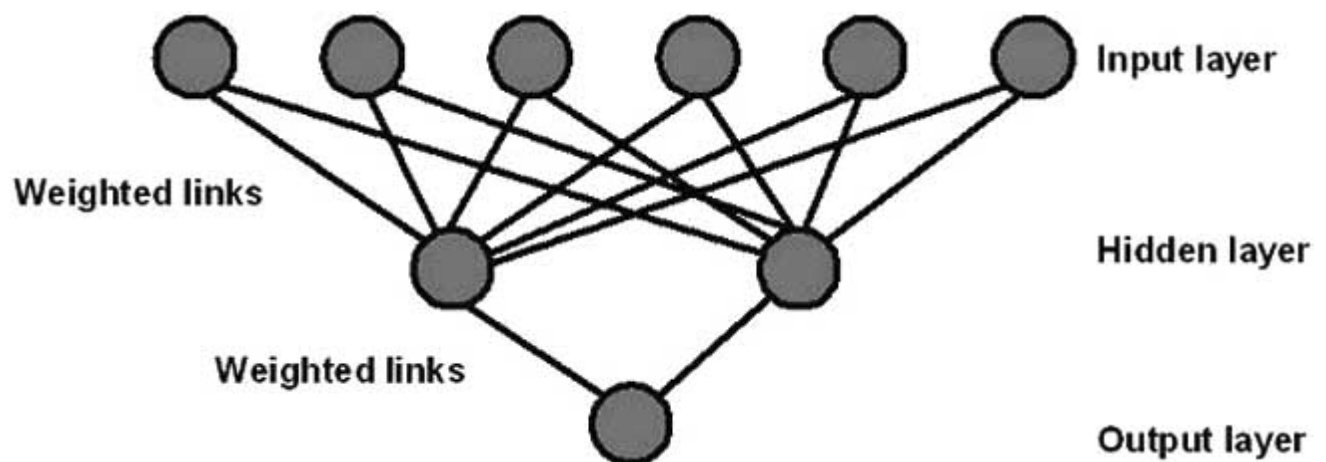


Fig. (1). Structure of the multi-layer perceptron artificial neural network. Each node in each layer is connected to each node in the next layer by interconnecting weighted links.

Hajmeer, 2000). These steps are repeated until a minimum error value is reached.

2. DEVELOPMENT OF ANN PROTOCOLS FOR PROTEOMIC SCREENING

The careful adjustment of the particular parameters to be used when developing an ANN model is a key issue in order to create a model which will generalise well on the data to be analysed. For the analysis of proteomic data, there are essentially six main parameters which need to be carefully selected before training of the ANN begins, these will now be discussed in more detail and are as follows; (i) partitioning of data, (ii) initial values of interconnecting weights, (iii) learning rate, (iv) momentum value, (v) training convergence criteria and (vi) the hidden layer size.

The quality of the results obtained using ANN models is highly dependant on the quality (and to a certain extent the size) of the data set used in model building, and therefore data which is used in the training process should be large enough to be representative of any variation contained within the problem. During the development of the ANN model, the data should be split into three distinct subsets; training data, test data, and validation data. The training data should constitute the majority of the data and be representative of the system as a whole as this data set is used during the learning process to update the weights in the network. During training, the ANN model is trained with the training data, and continually optimised against the test data set so that the architecture can be assessed by using the error values generated during training on the test data. The validation data set is used to assess the ANN model on unseen data once the model has been developed. Various percentage splits have previously been proposed and interesting results have been found. Bourquin *et al.*, (1998) used a data split of 68 samples for training, 12 for test purposes and 22 for validation, while Manel *et al.*, (1999) used 80 % of the data for training and the remaining 20 % for test. Khan *et al.*, (2001) and Jerez-Aragones *et al.*, (2003) preferred to split the data into $\frac{2}{3}$ for training and $\frac{1}{3}$ for validation, but from personal experience in our studies multiple splits of 60 % training, 20 % test and 20 % validation appears to be effective in ensuring that training reaches convergence and does not become stuck in any local minima. Multiple random sample cross validation also ensures that the performance accuracies of the models are not an artefact of splitting the data in certain ways, ensuring generalized models are obtained.

As the training algorithm begins initialising by randomising the weights in the network, a suitable choice of range for these weights is potentially important in allowing the network to train to produce a good set of weights, and may lead to improvements in the quality of training. Initial weight values should be small so that the sigmoidal transfer functions are not driven into saturation regions where a flat error surface would result, however if these are too small the sigmoidal activation functions will be approximately linear (Bishop, 1995). Studies by Kim and Park (2001) reveal that for their data, model performance was constantly degraded with an increase in initial weight distribution, while Lou and Perez (1996) claimed that the rate of model convergence is seriously affected by the initial weight parameter. Kuo

(2001) proposed a different approach which involved using ANNs with initial weights generated by a genetic algorithm. However, our studies showed that by using a range of initial weight ranges between 0 and 0.001, to 0 and 10, the mean squared error (MSE) and classification performance of the model on the validation data set was not significantly different between each initial weight value, therefore an arbitrary value of 0.001 is generally used.

The weight change of a neuron is proportional to the influence an input had on the error during training and the learning rate is a constant which controls the size of these weight changes. Each time a pattern is presented to the network, the weights leading to an output node are modified slightly during learning in the direction required to result in a smaller error the next time the same pattern is presented. The amount of weight modification is the learning rate times the error. For example, if the learning rate is 0.5, the weight change is one half the error value (Ward Systems Group, Inc., 1993). The larger the learning rate, the larger the weight changes, and the faster the learning will proceed. If the learning rate is too small, training will be slowed down, however, oscillation or non-convergence can occur if the learning rate is too large. Maier and Dandy (1998) found that when a lower learning rate was used (0.005), the error decreased slowly until a local minimum in the error was reached, and it remained there. When a larger learning rate was used (0.1), better predictions were found with faster training, as the network is more likely to escape from areas of local minima, and find a more global solution to the problem. These findings are comparable to those found in our studies, where learning rates between 0.1 and 1.0 were tested, with 0.1 producing the lowest MSE. This is because the learning rate values above 0.1 were causing the model to take steps that were too large, thus skipping the global minima so that convergence could not occur.

As mentioned, large learning rates may often lead to oscillation of weight changes which results in either the learning process never completing, or the model converging towards a non-optimal solution. One technique for a faster convergence towards minimum error, while also smoothing out the oscillations that may occur with a high learning rate, is to add a momentum factor to the BP learning algorithm. This momentum factor speeds up the training process by adding a proportion of the previous weight changes to the current weight changes. A high momentum will reduce the risk of the network being stuck in local minima, but risks overshooting the solution, while a small momentum leads to slower training (Basheer and Hajmeer, 2000). Generally a trial and error procedure is used when seeking out the best learning rate and momentum combinations for any particular data. Attoh-Okine (1999) found that a learning rate in the region of 0.2 to 0.5 with a momentum factor of 0.4 to 0.5 appeared to be the appropriate combination, while Maier and Dandy (1998) came to the conclusion that although altering the momentum factor had no significant effect on the error, the learning count was greatly reduced with a higher momentum (0.9). Raimundo and Narayanaswamy (2003) found several combinations of learning rate and momentum to be efficient during their study, with varying values of learning rates of between 0.05 and 0.4 being combined with a momentum value of either 0.3 or 0.5 to produce the most

effective model. In two separate studies, Mittal and Zhang (2000a; 2000b) tested various combinations of learning rates and momentum factors and found that in one study a learning rate of 0.7 and a momentum of 0.5 achieved the best prediction results, while in another, setting both the learning rate and momentum factor to 0.7 was more effective, showing that different values are appropriate for different data sets. Several momentum factors were tested in this study, with a value of 0.5 resulting in the lowest error for this data, so this was combined with a learning rate of 0.1.

Training convergence can be determined by a number of factors, depending on whether the training or the test set will be used as an indicator of when training should be stopped. The training set criteria (computed at the end of each epoch) which may be used are; (i) average error below a predefined level, (ii) epochs since the minimum average error exceeding a specified number, (iii) largest error below a predefined level and, (iv) learning epochs exceeding a predefined number. The test set criteria are computed at the end of each calibration interval, which is a defined number of epochs. These are, (i) average error below a predefined level, (ii) number of events since the minimum error exceeded a specified number and, (iii) largest error below a predefined level. The error on training data generally decreases with increasing number of epochs, with an initial large drop in error which slows down as the network begins to learn the patterns representing the data set. However, if training is allowed to continue beyond the point at which the error reaches the global minima, overfitting (or overtraining) may arise, where memorization of the training data occurs. Because of this overfitting, if a network performance is monitored by training data alone, the network will perform with little error on the training data but will not be able to generalise well for new data. For this reason, in this study model convergence was determined by a failure of the model to improve on the minimum MSE on the test data for 20,000 training events. This may be enhanced further by using multiple models with different unseen data sets (known as bootstrapping). This bootstrapping approach uses a large number of different training/test/validation data splits, thus creating several sub-models so that all data points are treated as unseen data a number of times. This repetition allows confidence intervals to be determined for any given model.

The purpose of the hidden layer is to enable the ANN to classify input data with non-linear characteristics. A network with too little hidden nodes in this layer will result in only a linear estimate of the solution to the problem due to fewer feature detectors, while too many hidden nodes will result in over-training where the model will follow the noise in the data, resulting in poor generalisation (Basheer and Hajmeer, 2000) and increased training times. Generally a trial and error approach is used in order to find the optimum number of hidden nodes, where a low number of hidden nodes are used to begin with and a stepwise approach is undertaken, gradually increasing the node size until a minimum error is reached, as shown by Srećnik *et al.*, (2002). Other heuristics have also been proposed for determining the number of hidden nodes, such as using the $2n + 1$ rule where n is the number of nodes in the input layer (Fletcher and Goss, 1993). One technique which may be used in order to opti-

mise the number of hidden nodes by using a more intelligent means is correlated activity pruning (CAPing) (Roadknight *et al.*, 2001). This approach begins with a large number of hidden nodes, and removes units with constant outputs over all of the training patterns (as these are not contributing in the solution). Hidden nodes with identical or opposite activation energies for all patterns can be combined, thus reducing the number of units until the optimum number is found. The process involves monitoring activation strengths at each hidden node and calculating the correlation coefficient for each pair of hidden nodes. The pair of nodes with the correlation coefficient nearest 1 (or -1) is replaced by one node and the weights and biases are consequently changed. This process is then repeated, until generalisation is lost and therefore the minimum number of hidden nodes can be found while maintaining good generalisation. This CAPing approach was utilised in this study in order to deduce that two hidden nodes were most favourable for the data to be modelled.

3. PARAMETERISATION

Once the model for data classification has been created, the next phase in finding a solution to the problem is parameterisation, which asks two questions:

1. What is important in the system we are modelling?
2. What can be removed?

Model parameterisation allows for the identification of inputs (in this case, m/z , i.e. mass:charge ratio values generated from the SELDI-MS) which have the most influence on correct sample classification. Therefore by determining which inputs are the most important in the model, non-contributing and “noisy” variables may be removed from the system, reducing the model complexity while increasing classification performance. This process may be split into several stages, which are; (i) data pre-screen and split, (ii) weightings analysis, (iii) stepwise additive approach, (iv) analysis of interactions between inputs and, (v) class extension.

3.1. Initial Data Split and Pre-Screen

Data, when exported in its raw form from the SELDI-MS, may contain a great deal of noise and redundancy. For this reason, data within the region of 0-3 kDa and anything greater than 30 kDa is removed prior to ANN analysis as this is deemed to be noisy and not representative of the system as a whole. To enhance this further, a cluster analysis may be performed to identify any strong outliers within the data set which may affect the ANN learning process. Cluster analysis includes a number of different classification algorithms which may be used to develop taxonomical relationships between samples as part of an exploratory data analysis and therefore may be used as a data pre-screening tool to remove any strong outliers which may cause a conflict in the data thus affecting ANN training.

As the data extrapolated from the SELDI-MS analysis is of such high dimensionality, the number of inputs remaining after this initial pre-screen are still in the region of 13,000. This would result in a model with long training times that

would be incapable of generalising well because of its complexity, so therefore the inputs may be reduced to just those which are contributing the greatest in classification.

3.2. Weightings Analysis

In order to reduce the number of inputs the data may be screened further by splitting the data into blocks and training each of these individually. In this study, a "rolling approach" was utilised which involves training data containing inputs spanning a 3 kDa mass range, and then shifting this along 1 kDa at a time, in order to create a new data block. For example, the first block contained data within the 3-6 kDa mass range, the next block ranged from 4-7 kDa, the next from 5-8 kDa and so on up to 30 kDa. Each model was then trained over 50 random training/test/validation subsets (bootstrapping) and relative importance values for each individual input was recorded in order to rank these inputs according to their influence upon correct sample assignment. These relative importance values are calculated by conducting a weightings analysis of the trained network, and taking the sum of the absolute weight values leading from each input to the output. After each model was trained, a proteomic profile could be produced showing the relative importance of inputs over the mass range of 3-30 kDa.

From this relative importance analysis, inputs with the greatest importance were selected from the data in order to reduce the number of inputs in the model. This was achieved by selecting the top 1,000 inputs with the greatest relative importance values and repeating the training process as described previously. Relative importance analysis was again used to determine the top 100 inputs from these 1,000. This was repeated to deduce the top 70, 60, then finally the top 50 inputs from the initial data set of approximately 13,000 in terms of relative importance. Therefore this approach of analysing the weights of the trained network can be used as an effective means to greatly reduce the number of inputs in a model in order to decrease model complexity, while increasing predictive capabilities as a result of noise removal.

3.3. Stepwise Additive Approach

Once the data has been parameterised in order to leave only the most important elements as inputs, the next stage is to identify the minimum number of inputs which are required to correctly assign the samples into their respective group, thus identifying key protein molecular ions observed in SELDI that are representative of specific treatment regimens. This was achieved using a stepwise additive approach which involves training a number of different models using the 50 inputs identified during the weightings analysis. Each input was used as a single input in the model, and for each model, 100 random training/test/validation bootstrapped subsets were used in order to provide a measure of confidence in the predictions made. The MSE was calculated, and the model with the lowest value was selected for further training. All of the remaining inputs were then added sequentially to this first input creating 49 two-input models and these were trained as before with 100 random training/test/validation subsets. The model with the best performance was selected to produce a three-input model, then a four-

input model, and so on until it was deemed that model convergence was reached with an 8 input model.

4. RESULTS FROM BREAST CANCER STUDY

So far this manuscript has discussed the importance of the main parameters regarding the development of ANN architecture which need to be adjusted prior to ANN analysis, and how to parameterise the models to create robust models containing only inputs of a high importance in classification. This section will now discuss the results found when applying these approaches to proteomic data generated from SELDI-MS profiling of cell lines from patients with breast cancer, with the aim of identifying proteomic markers representative of treatment regimen.

4.1. Model Parameters

This study utilised a three layer MLP ANN with a BP algorithm to model for 96 breast cancer cell lines, 48 of which were untreated (control) samples, and the remaining 48 were chemosensitive (MCF-7 and T47-D) cell lines, treated with the drug paclitaxel, which was removed at differing time periods at either 1, 2 or 3 days (24, 48 or 72 hours respectively). An arbitrary value of 0.001 was selected for the initial weights since by using a range of initial weightings no significant difference was evident by comparing the MSE values for each model, as shown in Fig. (2).

As described earlier, the careful selection of the right combination of learning rate and momentum factor values is of paramount importance when creating an ANN model. If the user is not vigilant in this selection, the network may become stuck in local minima, train slowly or oscillation of weight changes may occur, all of which leads to poor generalisation on new unseen data. In this study architectures with learning rate and momentum values between 0.1 and 0.9 were trained in order to deduce the ANN model which performed best for this data. Using the MSE value as a means of measuring prediction accuracy, it was found that a learning rate of 0.1 with a momentum value of 0.5 produced the lowest value as seen in Fig. (3).

In order to deduce the number of hidden nodes the hidden layer contained, a CAPing approach was used. One thousand hidden nodes were initially used and the model was trained. Activation strengths at each node were examined and pairs of nodes with correlation coefficients nearest 1 or -1 were removed, resulting in 16 nodes remaining. The procedure was then repeated for these 16 nodes, which removed 8 of these which were then retrained resulting in 2 hidden nodes. Fig. (4) shows that this model containing 2 hidden nodes was shown to be the minimum number of nodes to maintain good generalisation on validation data and therefore this number was used for the remainder of the study.

4.2. Data Parameterisation

An initial cluster analysis of the data was performed. Although this (because of its linear approach) cannot distinguish between the different groups, it is able to highlight any strong outliers in the data which may be removed pre-ANN analysis in order to prevent any inconsistencies which may affect the training process. Fig. (5) illustrates results

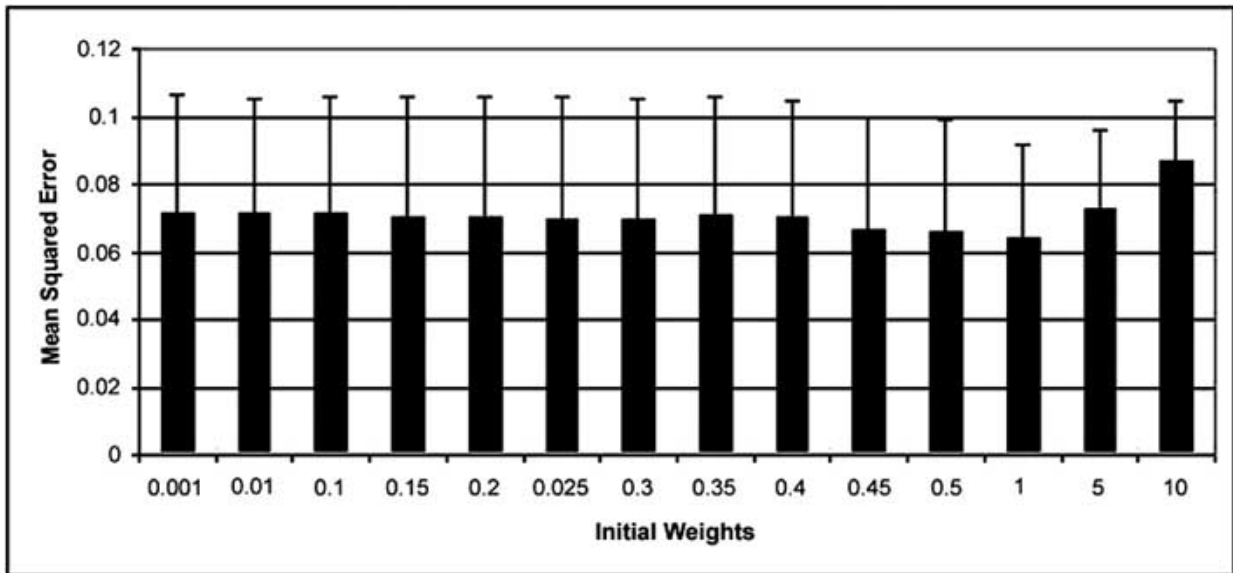


Fig. (2). Initial Weights. A range of initial weights were tested in order to deduce if this value had an effect on model performance.

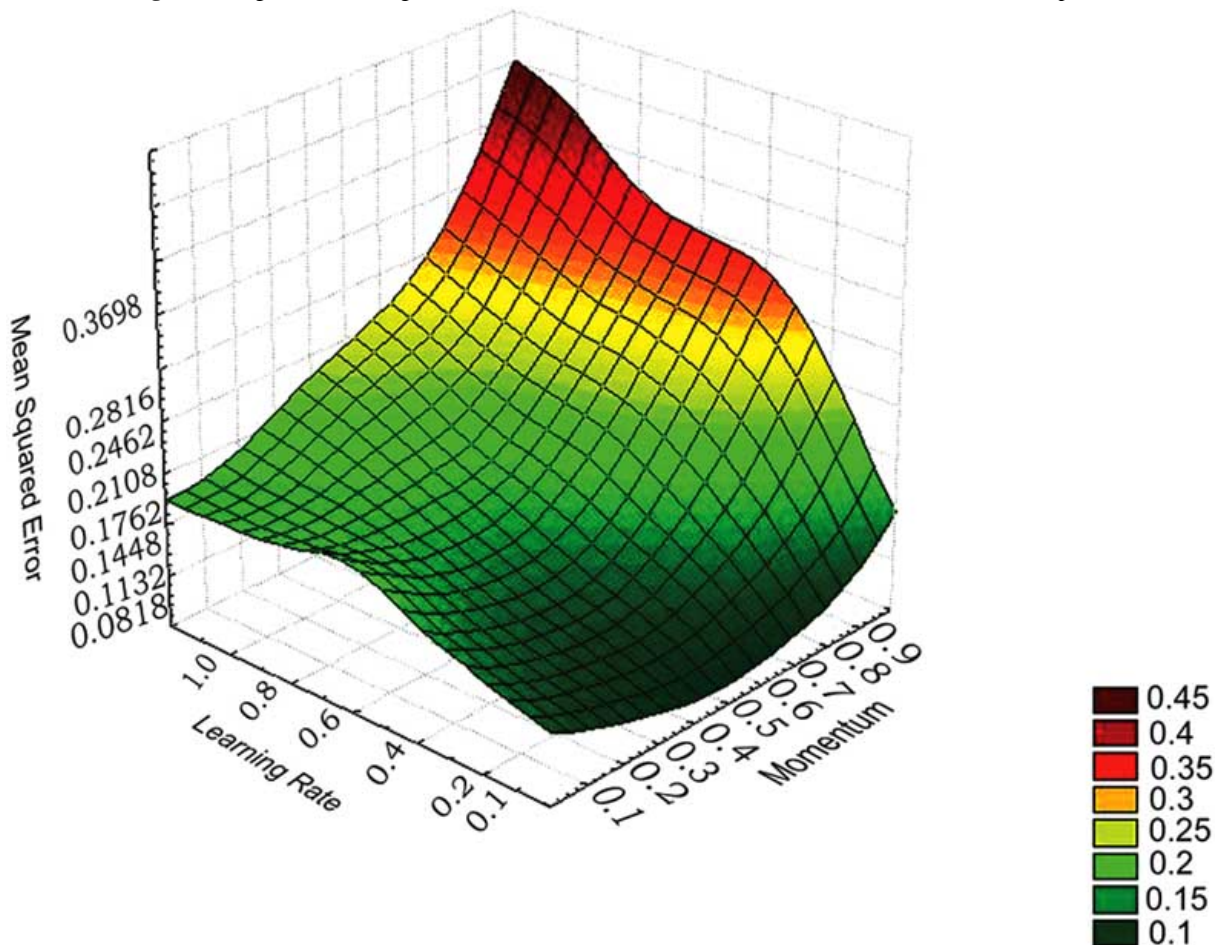


Fig. (3). Optimisation of learning rate and momentum factor values. Architectures with various momentum factor values and learning rates were trained in order to deduce the optimal ANN model for the data.

from the cluster analysis which showed that none of the samples were causing any such conflict, therefore all of the samples were used in training.

The data was then split into blocks containing inputs over a 3 kDa mass range using the rolling approach as described previously, showed diagrammatically in Fig. (6). In order to

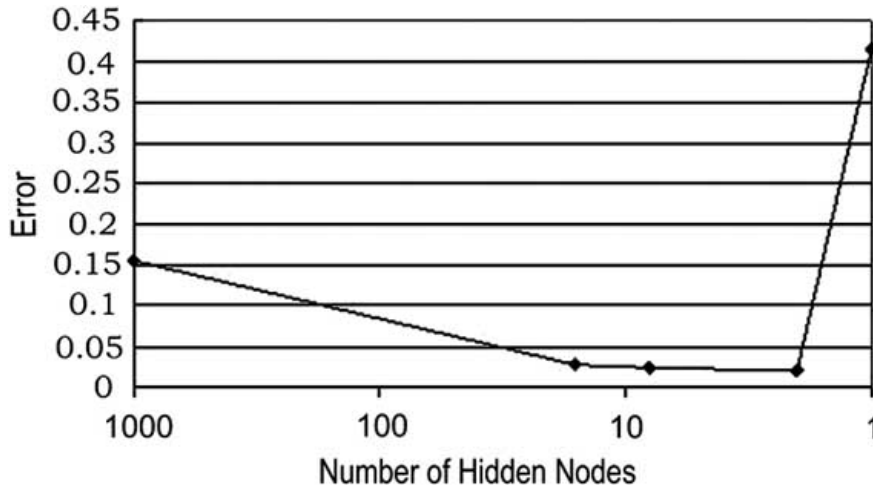


Fig. (4). Hidden nodes. A CAPing approach was used to determine the minimum number of hidden nodes that could be used while maintaining good generalisation. The number of hidden nodes are presented on a logarithmic scale, with points representing one thousand, sixteen, eight, two and one hidden nodes.

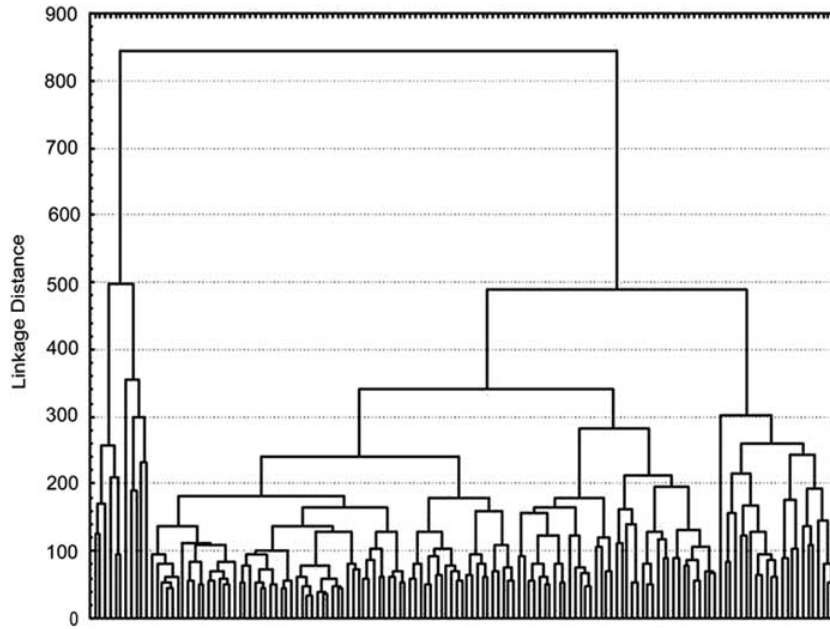


Fig. (5). Cluster analysis. Tree diagram from cluster analysis showing complete linkage with regard to Euclidean distances for 96 variables.

train the models, the data was split into 3 sets. The first set, which contained 60 % of the data was used for training, a

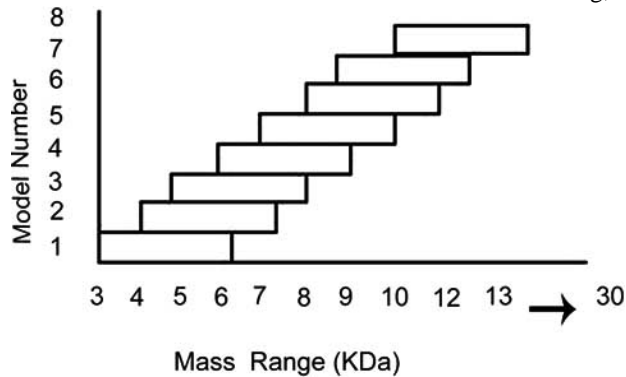


Fig. (6). Rolling approach. Data was split into blocks spanning 3 kDa mass range and shifted along 1 kDa for each new block created.

further 20 % was used for test purposes during training, and the final set was the remaining 20 % which was set aside for validation of the models created. Each block was trained over multiple random training/test/validation subsets, known as bootstrapping. This approach ensures that each sample is used as validation data a number of times, allowing for the calculation of levels of confidence for each given model created. Once each model had been trained, a weightings analysis was conducted and inputs were ranked according to their relative importance values enabling a proteomic profile of the whole mass range to be produced, as seen in Fig. (7).

This profile was used to identify the top 1,000 inputs in the model which were then trained as before. These inputs were then ranked in order of high to low relative importance (seen in Fig. (8)) in order to determine the top 100 most important inputs from these 1,000. This was repeated to establish the top 70, 60, and finally the top 50 inputs which were necessary for the accurate classification of treatment

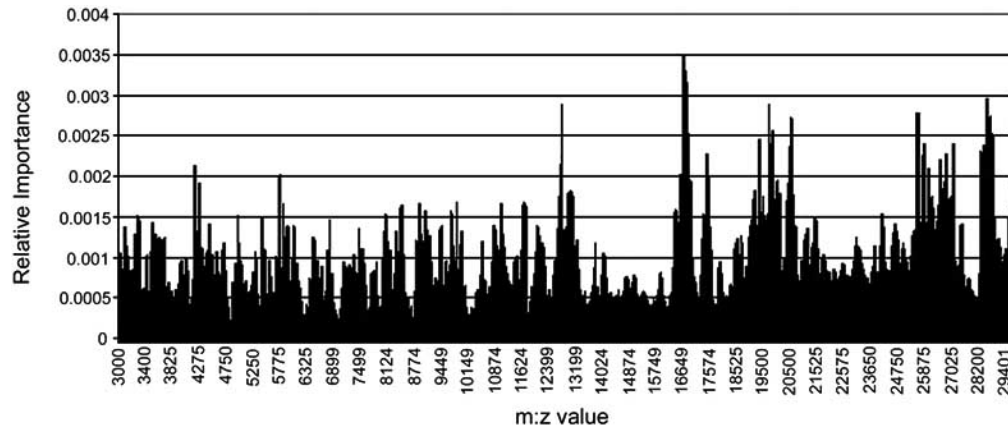


Fig. (7). Proteomic profile. Relative importance values for ion masses ranging from 3,000-29,994 Da. These values illustrate a value obtained from multiple sub-models of each individual model in which different random weightings were applied to each.

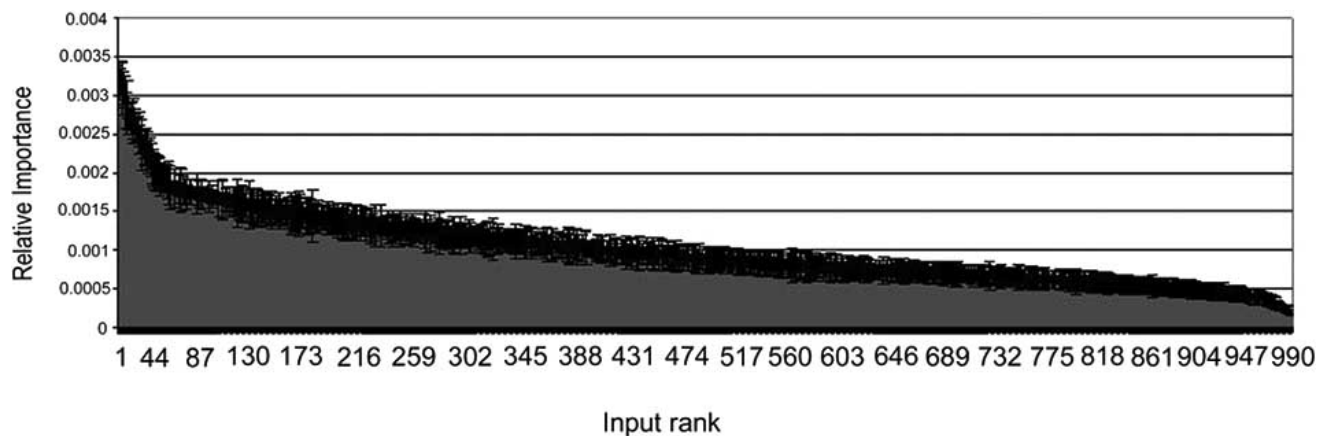


Fig. (8). Relative importance profile for the 1,000 most important inputs. Values were ranked in descending order from input of most importance to input of least importance. Ninety five percent confidence interval error bars are shown in black.

type, as seen in Table (1). Note that the majority of these 50 inputs appear to arise in clusters around certain mass values which is interesting as when data is imported into the ANN model, each data point is treated as an independent variable with no relationship between that and any other input.

4.3. Additive Approach

Once these 50 inputs had been identified, the next stage was to determine the minimum number of protein/peptide molecular ions capable of making accurate predictions, to discern which combination of ions would act as the most accurate markers indicative of treatment, and consequently may correlate to this. To achieve this, a stepwise additive approach was implemented as described in section 3.3. This initially involved using each ion individually and assessing its performance using the MSE value generated. The ion which obtained the lowest MSE value was then combined into a two-ion model with each of the remaining 49 ions. The combination of ions producing the lowest MSE was then

used in conjunction with the remaining 48 ions to produce a three-ion model and so on. This procedure continued until an 8 ion model was reached and the model which performed with the most accuracy could be found, thus determining the ions which had the greatest influence in classification. Fig. (9) shows the MSE and the accuracy as a percentage of samples classified correctly for all 8 models. An output of 1 would signify an untreated sample and an output of 2 would signify a sample treated with paclitaxel, with a error of prediction value of 0.5 being the threshold for determining network prediction. It is evident from Fig. (9) that with each ion that is sequentially added to the model, the MSE showed an obvious decrease with the minimum value being 0.042. Similarly, with each ion added to each model, the classification accuracy increased until the 6 ion model, where it reached its peak accuracy of 94.8 %. The additive approach was then continued to ensure model convergence, which was deemed to occur at the 8 input model. Fig. (10) shows the mean network error values for each individual sample.

Table 1. Fifty Molecular Masses With the Highest Relative Importance Values, Grouped into Their Respective Clusters

Input m/z value	Relative Importance	Input m/z value	Relative Importance
3082	0.0233	8060	0.0118
3267	0.0157	8061	0.0126
3268	0.0208	8078	0.0131
3314	0.0132	8080	0.0143
3315	0.0127	8081	0.0141
4189	0.0221	8902	0.0172
4190	0.0135	8904	0.0180
4274	0.0217	8906	0.0202
4275	0.0323	8908	0.0188
4954	0.0167	8909	0.0195
5415	0.0294	9851	0.0127
5768	0.0198	9853	0.0133
5769	0.0239	9855	0.0117
5770	0.0217	11042	0.0143
5772	0.0203	11044	0.0147
5773	0.0199	11046	0.0131
5775	0.0182	11084	0.0134
5846	0.0138	11720	0.0104
5848	0.0160	13077	0.0119
5849	0.0158	13079	0.0115
5914	0.0158	16711	0.0109
5915	0.0213	16714	0.0111
5924	0.0219	16728	0.0129
6823	0.0268	19675	0.0150
7471	0.0108	26438	0.0128

Model performance was then assessed using a Receiver Operating Characteristic (ROC) curve. A ROC curve determines the number of true positives (or correctly defined paclitaxel treated samples), true negatives (correctly defined control samples), false positives (incorrectly defined paclitaxel treated samples) and false negatives (incorrectly defined control samples) and defines a summary statistic for performance. It achieves this by plotting the true positive rate against the false positive rate at different possible cutpoints (in this case, prediction error thresholds). ROC curves for the one, three, five and eight-ion models can be seen in Fig. (11), and results are summarised in Table (2) to show a clear representation of how the model performance changes with the addition of more ions. It is clear from Table (2) that a ROC curve provides information about several different variables. To explain these briefly, accuracy is the overall ability of the model to correctly assign the samples.

The sensitivity is the percentage of the paclitaxel treated samples correctly classified while the specificity is the percentage of control samples correctly classified. The positive predictive value shows the percentage of the true positives distinguished from the false positives and the negative predictive value is the percentage of true negatives from false negatives. The area under the curve (AUC) measures discrimination, which is, the ability of the model to correctly classify between control and drug treated samples. A perfect ROC curve (and therefore a perfect test) has an AUC value of 1, so the closer the curve follows the left hand border and then the top border of the ROC space, the more accurate the test is. From the results, it is clear that when the number of inputs was increased, the accuracy of the model also increased until it reached a point where it appeared to become stable. The best model contained 8 ions and had an AUC value of 0.993 which signifies a very good test, with

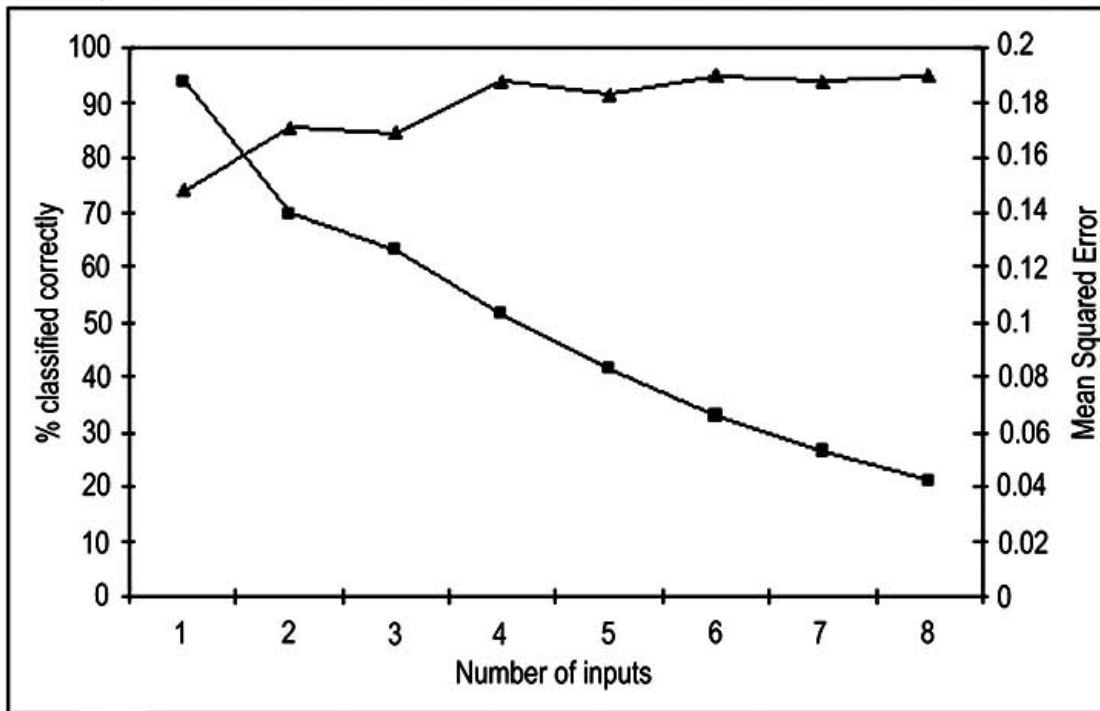


Fig. (9). Chart shows both the mean squared error (■) and % of samples classified correctly (▲) for the 8 models developed using an additive approach.

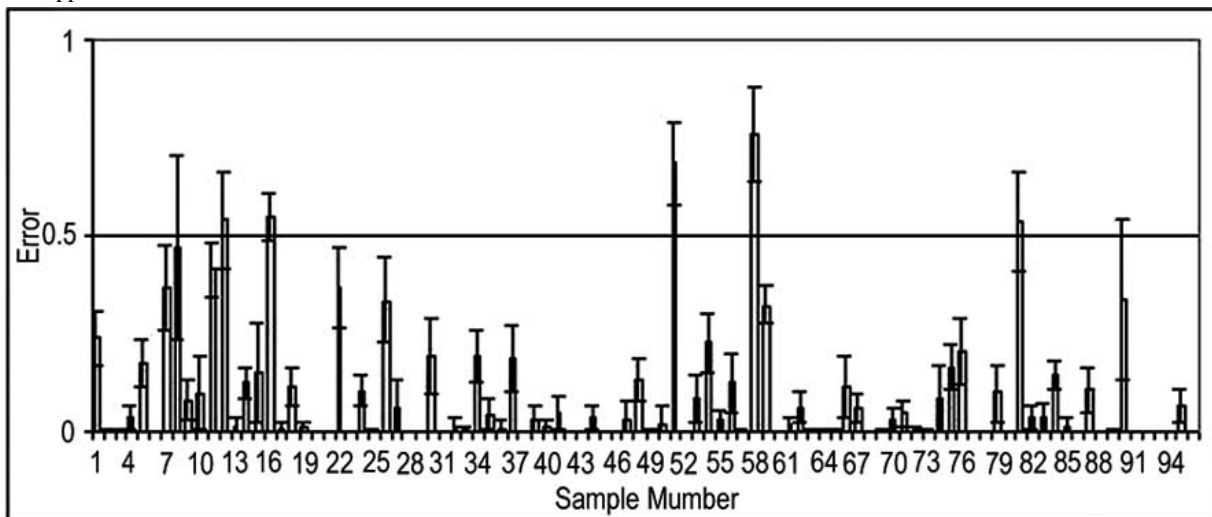


Fig. (10). Shows network error for all 96 samples for 8 input model, where an error greater than 0.5 symbolises an incorrect classification. Error bars represent 95 % confidence intervals.

low occurrences of false positives and false negatives. This model performed with an accuracy of 94.8 % classifying 91 out of the 96 samples into their respective groups with sensitivity and specificity also high (93.8 % and 95.8 % respectively).

4.4. Relative Importance Values for Top 8 Inputs

The top 8 inputs were then analysed to determine if their respective relative importance values showed a marked change as the cell lines were exposed to paclitaxel over increasing time periods, and subsequently could be potential markers indicative of treatment regimen. Fig. (12) shows that

the majority of the ions showed an increase or reduction in intensity as drug exposure increased in time compared to the untreated control samples. In particular ion 8909 showed an increase in intensity between the untreated samples and those treated at day 1, ions 4274 and 13077 showed a significant increase in intensity between days 1 and 2, as did ions 3082 and 5924 between days 2 and 3. A clear decrease in intensity was evident for ion 16728 throughout the time period, with a significant decrease after day 2, and this decrease in intensity between days 2 and 3 was also statistically significant in ion 8909. These results may be representing a true up or down regulation in ion concentration of these ions, which may be confirmed with further studies.

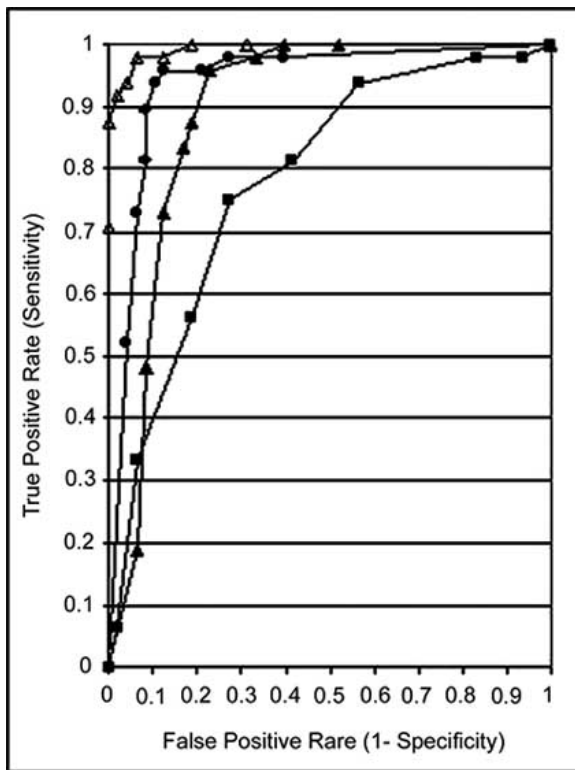


Fig. (11). Diagram of ROC curves for 1-ion model (■), 3 ion-model (▲), 5-ion model (●) and 8-ion model (△).

5. DISCUSSION

In order to allow for the complex analysis of the vast amounts of data generated from SELDI-MS analysis (approximately 1.2 million data points contained within a data set of 96 samples), robust computer algorithms need to be developed. ANNs are a powerful tool which may provide a solution to this problem, as they are able to cope with huge data sets while also allowing for background noise and measurement inaccuracies. The steps and considerations that need to be taken in order to generate a model capable of high classification accuracies, and a model that is also generalised well enough to give a real world solution to the problem, have been discussed in this manuscript. In addition to this, advanced methodologies such as parameterisation and additive approaches have also been described, in conjunction with results from a study using these techniques in order to identify ions capable of classifying samples containing different breast cancer cell lines, which may be candidate

biomarkers representative of different treatment regimens. Such biomarkers could have significant effects on the treatment and management of patients with certain diseases, such as cancer.

Using a rolling approach in order to parameterise the data, an initial input set of approximately 13,000 inputs was reduced to just 8 inputs of highest relative importance in terms of correct sample classification. This was achieved by using a bootstrapping procedure as a means of treating each sample as unseen data a number of times, in order for confidence intervals to be calculated. These 8 inputs were then used as part of a stepwise additive approach with the aim of identifying the minimum number of inputs required to accurately classify the data, and to determine whether combinations of these inputs would enhance model performance further. As the number of inputs were added to the model, classification performance increased until the 6 ion model, which classified 94.8 % of the samples correctly. ROC curves were also generated for each model which showed a clear improvement in performance as more ions were added to the system. The 8 ion model showed an AUC value of 0.993, suggesting that these 8 ions are contributing greatest in prediction of the different treatments, and are acting as the most accurate biomarkers for this particular model. These top 8 inputs were also analysed for any change in relative importance as the samples are exposed to drug treatment over increasing time periods and there was a clear up or down regulation in intensity in several inputs, which may represent a true increase or decrease in ion concentration.

The predictive capabilities of the models generated in this study are comparable to those discussed by Petricoin *et al.*, (2002) where a high sensitivity and specificity was found using GAs and self-organising cluster analysis in the analysis of proteomic patterns in serum to identify ovarian cancer. The methods employed in our study generate models with predictive capabilities that compare favourably to accuracies shown by Batuello *et al.*, (2001) who used an ANN model to assess lymph node spread in patients with clinically localized prostate cancer where the authors reported AUC values of 0.81 and 0.77. Yang *et al.*, (2002) showed ANNs to be a better technique than principal components analysis in the classification of healthy persons and cancer patients using nucleoside data, where they reported classification accuracies of 95.8 and 92.9 % on validation data, although only a relatively small sample size was used in this study. Chen *et al.*, (2002) utilised a MLP ANN with a BP algorithm for the differential diagnosis of breast tumours on sonograms. Again, predictive accuracies here were comparable to those

Table 2. Comparison of Performances for Various Models

Ions in model	Accuracy	Sensitivity	Specificity	Positive Predictive Value	Negative Predictive Value	AUC
1	79.96	75	72.92	73.47	74.47	0.785
3	84.38	87.5	81.25	82.35	86.67	0.893
5	91.67	93.75	89.59	90	93.48	0.939
8	94.79	93.75	95.83	95.74	93.88	0.993

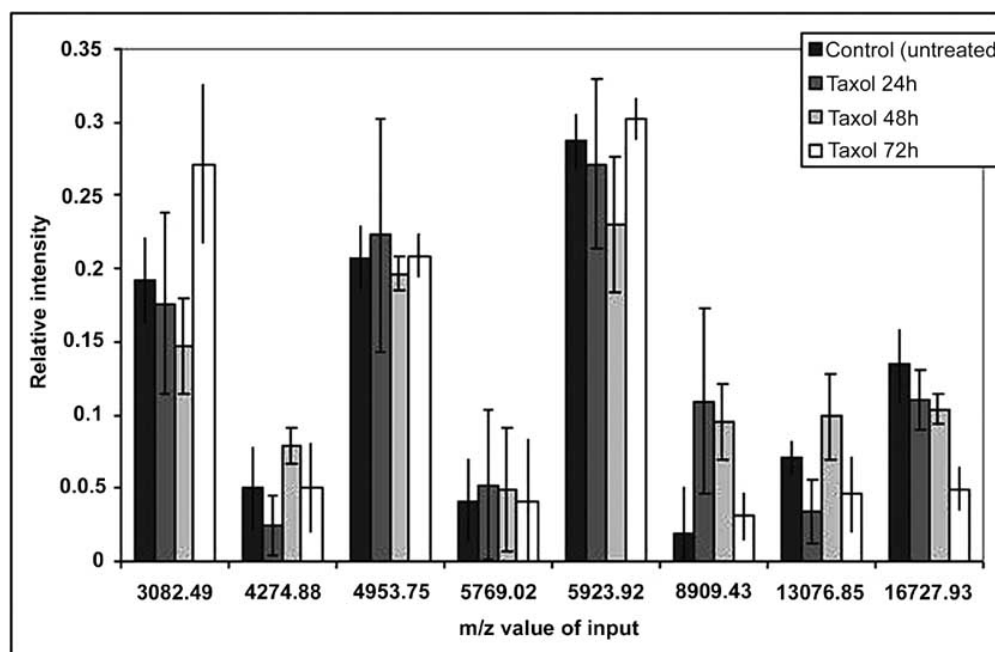


Fig. (12). Relative importance values for top 8 ions as treatment time increases from 24-72 hours.

in our study where a value of 0.9396 was reported when evaluating the model with a ROC curve. The ANN methodologies developed in our study are shown to be analogous to performance accuracies found by Chou *et al.*, (2001) when using stepwise logistic regression in the analysis of breast tumours, where the model developed by these authors classified to an accuracy of 91 %. The optimization methods developed in this study are also showing high accuracies when compared to those shown by Douglas *et al.*, (1996) in their studies, who applied ANNs as a staging tool with a sensitivity of 94 % and specificity of 69 %, while similarly Tewari and Narayan (1998) also used ANNs in this manner and developed a model with a sensitivity of 81-100 % with a specificity of 72-75 %. Finally, Khan *et al.*, (2001) proposed a novel method of diagnosis of cancers using gene expression profiling and ANNs. The authors concluded that this ANN approach offered “an alternative and powerful technique for the detection of gene expression signatures, and the discovery of novel genes that characterize a diagnostic subgroup may also identify new targets for therapy”.

CONCLUSION

Taking into account all of the parameters necessary in ANN model building, we have developed novel methodologies to create a model capable of classifying breast cancer cell lines into either control or drug treated groups using data derived from SELDI-MS analysis. This model consists of 8 inputs from an initial 13,000 that performs with an accuracy of almost 95 % on unseen validation data which shows that although a relatively small data set was used, the model has generalised well enough to be representative of a “real-world” solution. Several ions have also been identified which have a relative intensity that is either up or down regulated as the length of drug treatment is increased, which

may form the basis of future studies in order to conclude whether this is a true representation of whether an increase or decrease in ion concentration is occurring. These results show that ANNs are powerful enough to model for complex biological data which may lead to the identification of biomarkers which correlate to different treatments, or molecules which may have a role in chemosensitivity and whose expressions may be associated with specific biological states. This may form the basis for the development of models capable of predicting clinical outcomes, or a patient’s response to a particular treatment, which may be fundamental in the determination of diagnostic markers.

ABBREVIATIONS

AI	= Artificial Intelligence
ANN	= Artificial Neural Network
ART	= Adaptive Resonance Theory
AUC	= Area under the curve
BP	= Back Propagation
CAPing	= Correlated Activity Pruning
GA	= Genetic Algorithms
MLP	= Multi Layer Perceptron
MS	= Mass Spectrometry
MSE	= Mean Squared Error
m/z	= Mass to charge ratio
RBF	= Radial Basis Function
RNN	= Recurrent Neural Network
SELDI	= Surface Enhanced Laser Desorption/Ionisation

REFERENCES

- Andrews, S.G. (2003). Novel neural network methods for describing attributes contained within lesions images. 17th European Simulation Multiconference, Nottingham, pp. 142-7
- Attoh-Okin, N.I. (1999). Analysis of learning rate and momentum term in backpropagation neural network algorithm trained to predict pavement performance. *Adv. Eng. Software* **30**: 291-302.
- Ball, G., Mian, S., Holding, F., Allibone, R.O., Lowe, J., Ali, S., Li, G., McCardle, S., et al. (2002). An integrated approach utilizing artificial neural networks and SELDI mass spectrometry for the classification of human tumours and rapid identification of potential biomarkers. *Bioinformatics* **18**: 395-404.
- Balls, G.R., Palmer-Brown, D. and Sanders, G.E. (1996). Towards unravelling the complex interactions between microclimate ozone dose and ozone injury in clover. *Water Air Soil Pollut.* **85**: 1467-72.
- Barlow, J. (1995). Self-organizing maps and molecular similarity. *J. Mol. Graph* **13**: 25-7.
- Basheer, I.A. and Hajmeer, M. (2000). Artificial neural networks: fundamentals, computing, design, and application. *J. Microbiol. Methods* **43**: 3-31.
- Batuello, J.T., Gamito, E.J., Crawford, F., Han, M., Partin, A.W., McLeod, D.G. and O'Donnell, C. (2001). Artificial neural network model for the assessment of lymph node spread in patients with clinically localized prostate cancer. *Urology* **57**: 481-5.
- Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*, Oxford University Press Inc, New York.
- Bourquin, J., Schmidli, H., van Hoogevest, P. and Leuenberger, H. (1998). Advantages of Artificial Neural Networks (ANNs) as alternative modelling technique for data sets showing non-linear relationships using data from a galenic study on a solid dosage form. *Eur. J. Pharm. Sci.* **7**: 5-16.
- Boyle, M.D.P., Romer, T.G., Meeker, A.K. and Sledjeski, D.D. (2001). Use of surface-enhanced laser desorption/ionization protein chip system to analyze streptococcal exotoxin B activity secreted by *Streptococcus pyogenes*. *J. Microbiol. Methods* **46**: 87-97.
- Cao, Y. and Wu, J. (2002). Projective ART for clustering data sets in high dimensional spaces. *Neural Netw.* **15**: 105-20.
- Carpenter, G.A. and Grossberg, S. (1988). The ART of Adaptive Pattern Recognition by a Self-Organising Neural Network. *IEEE Computer* **21**: 77-88.
- Chen, D.R., Chang, R.F., Kuo, W.J., Chen, M.C. and Huang, Y.L. (2002). Diagnosis of breast tumours with sonographic texture analysis using wavelet transform and neural networks. *Ultrasound Med. Biol.* **28**: 1301-10.
- Chou, Y.H., Tiu, C.M., Hung, G.S., Wu, S.C., Chang, T.Y. and Chiang, H.K. (2001). Stepwise logistic regression analysis of tumour contour features for breast ultrasound diagnosis. *Ultrasound Med. Biol.* **27**: 1493-8.
- Cowper, M.R., Mulgrew, B. and Unsworth, C.P. (2002). Nonlinear prediction of chaotic signals using a normalised radial basis function network. *Signal Process* **82**: 775-89.
- Desilva, C.J.S., Choong, P.I. and Attikiouzel, Y. (1994). Artificial neural networks and breast-cancer prognosis. *Aust. Comput. J.* **26**: 78-81.
- Douglas, T., Connolly, R., McLeod, D., Moul, J. and Snow, P. (1996). Neural network analysis of pre-operative and post-operative variables to predict pathological stage and recurrence. *J. Urol.* **155**: 477A.
- Dreiseitl, S. and Ohno-Machado. (2002). Logistic regression and artificial neural network classification models: a methodology review. *J. Biomed. Inform.* **35**: 352-9.
- Fletcher, D. and Goss, E. (1993). Forecasting with neural networks : An application using bankruptcy data. *Information & Management* **24**: 159-67.
- Hecht-Nielsen, R. (1988). Applications of counterpropagation networks. *Neural Netw.* **1**: 131-9.
- Hopfield, J.J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Nat. Acad. Sci. USA* **79**: 2554-8.
- Jerez-Aragónés, J.M., Gómez-Ruiz, J.A., Ramos-Jiménez, G., Muñoz-Pérez, J. and Alba-Conejo, E. (2003). A combined neural network and decision trees model for prognosis of breast cancer relapse. *Artif. Intell. Med.* **27**: 45-63.
- Khan, J., Wei, J.S., Ringné, M., Saal, L.H., Ladanyi, M., Westermann, F., Berthold, F., Schwab, M., et al. (2001). Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nat. Med.* **7**: 673-9.
- Kim, B. and Park, S. (2001). An optimal neural network plasma model: a case study. *Chemom. Intell. Lab. Syst.* **56**: 39-50.
- Kim, Y.J., Kim, Y.K. and Cho, Y. (1998). A heuristic-based genetic algorithm for workload smoothing in assembly lines. *Computers Ops. Res.* **25**: 99-111.
- Kohonen, T. (1989). *Self-Organization and Associative Memory*. Springer Series in Information Sciences, Springer, Berlin, 3rd Edition.
- Kuo, R.J. (2001). A sales forecasting system based on fuzzy neural network with initial weights generated by genetic algorithm. *Eur. J. Oper. Res.* **129**: 496-517.
- Lancashire, L.J., Mian, S., Rees, R.C. and Ball, G.R. (2003). Preliminary artificial neural network analysis of SELDI mass spectrometry data for the classification of melanoma tissue, 17th European Simulation Multiconference, Nottingham, 131-35.
- Li, Y., J, J-H., C, Z-P., Xu, C-J. and Yu, R-Q. (1999). A new method based on counterpropagation network algorithm for chemical pattern recognition. *Anal. Chim. Acta.* **388**: 161-70.
- Lou, K-N. and Perez, R.A. (2001). A new system identification technique using Kalman filtering and multilayer neural networks. *Artif. Intell. Eng.* **10**: 1-6.
- Lucasius, C.B. and Kateman, G. (1993). Understanding and using genetic algorithms Part I. Concepts, properties and context. *Chemom. Intell. Lab. Syst.* **19**: 1-33.
- Maier, H.G. and Dandy, G.C. (1998). The effect of internal parameters and geometry on the performance of back-propagation neural networks: an empirical study. *Environ. Modell. Softw.* **13**: 193-209.
- Manel, S., D, J-M. and Ormerod, S.J. (1999). Comparing discriminant analysis, neural networks and logistic regression for predicting species distributions: a case study with a Himalayan river bird. *Ecol. Modell.* **120**: 337-47.
- Mian, S., Ball, G., Hornbuckle, J., Holding, F., Carmichael, J., Ellis, I., Ali, S., Li, G., et al. (2003). A prototype methodology combining surface-enhanced laser desorption/ionization protein chip technology and artificial neural network algorithms to predict the chemoresponsiveness of breast cancer cell lines exposed to Paclitaxel and Doxorubicin under *in vitro* condition. *Proteomics* **3**: 1725-37.
- Mittal, G.S. and Zhang, J. (2000a). Prediction of temperature and moisture content of frankfurters during thermal processing using neural network. *Meat. Sci.* **55**: 13-24.
- Mittal, G.S. and Zhang, J. (2000b). Prediction of freezing time for food products using a neural network. *Food Res. Intern.* **33**: 557-62.
- Moody, J. and Darken, C. (1989). Fast learning in networks of locally-tuned processing units. *Neural Comput.* **1**: 281-94.
- Niculescu, S.P. (2003). Artificial neural networks and genetic algorithms in QSAR. *J. Mol. Struct-Theochem.* **622**: 71-83.
- Nour, M.A. and Madey, G.R. (1996). Heuristic and optimization approaches to extending the Kohonen self organizing algorithm. *Eur. J. Oper. Res.* **93**: 428-48.
- Palmer-Brown, D., Wee Lee, S., Tepper, J. and Roadknight, C. (2003). Fast learning neural nets with adaptive learning styles. 17th European Simulation Multiconference, Nottingham, 118-23.
- Paweletz, C. P., Liotta, L. A. and Petricoin, E. F. (2001). New technologies for biomarker analysis of prostate cancer progression: Laser capture microdissection and tissue proteomics. *Urology* **57**: 160-3.
- Petricoin, E.F., Ardekani, A.M., Hitt, B.A., Levine, P.J., Fusaro, V.A., Steinberg, S.M., Mills, G.B., Simone, C., et al. (2002). Use of proteomic patterns in serum to identify ovarian cancer. *The Lancet* **359**: 572-7.
- Pritchard, L. and Dufton, M.J. (2000). Do proteins learn to evolve? The hopfield network as a basis for the understanding of protein evolution. *J. Theor. Biol.* **202**: 77-86.
- Raimundo, Jr. I.M. and Narayanaswamy, R. (2003). Simultaneous determination of Zn(II), Cd(II) and Hg(II) in water. *Sens Actuators B* **90**: 189-97.
- Roadknight, C., Palmer-Brown, D. and Al-Dabass, D. (2001). Simulation of correlation activity pruning methods to enhance transparency of ANNs. UKSIM2001, Conf. Proc. of the UK Simulation Society, Emmanuel College, Cambridge, 56-62.
- Rumelhart, D.E. and McClelland, J.L. (1986). *Parallel distribution processing: Explorations in the microstructure of cognition*, Volume 1: Foundations. MIT Press, Cambridge, MA, USA.
- Smith, K.A., Abramson, D. and Duke, D. (2003). Hopfield neural networks for timetabling: formulations, methods and comparative results. *Comp. Ind. Eng.* **44**: 283-305.

- Srecnik, G., Debeljak, Z., Cerjan-Stefanovic, S., Novic, M. and Bolanca, T. (2002). Optimization of artificial neural networks used for retention modelling in ion chromatography. *J. Chromatogr. A* **973**: 47-59.
- Tewari, A. and Narayan, P. (1998). Novel staging tool for localized prostate cancer: a pilot study using genetic adaptive neural networks. *J. Urol.* **160**: 430-6.
- Ultsch, A. and Röske, F. (2002). Self-organizing feature maps predicting sea levels. *Inf. Sci.* **144**: 91-125.
- Ward Systems Group, Inc., (1993). Neuroshell 2 Help. Ward Systems Group, Inc. Frederick, Maryland, (www.neuroshell.com).
- Wei, J.T., Zhang, Z., Barnhill, S.D., Madyastha, K.R., Zhang, H. and Oesterling, J.E. (1998). Understanding artificial neural networks and exploring their potential applications for the practising urologist. *Urology* **52**: 161-72.
- Yang, J., Xu, G., Kong, H., Zheng, Y., Pang, T. and Yang, Q. (2002). Artificial neural network classification based on high-performance liquid chromatography of urinary and serum nucleosides for the clinical diagnosis of cancer. *J. Chromatogr. B* **780**: 27-33.
- Zhukov, T.A., Johanson, R.A., Cantor, A.B., Clark, R.A. and Tockman, M.S. (2003). Discovery of distinct protein profiles specific for lung tumors and pre-malignant lung lesions by SELDI mass spectrometry. *Lung Cancer* **40**: 267-79.

Received: May 4, 2004

Accepted: February 8, 2005