Neural, Parallel, and Scientific Computations 24 (2016) 83-96

COMPARISON OF ACTIVATION FUNCTIONS IN MULTILAYER NEURAL NETWORKS FOR STAGE CLASSIFICATION IN BREAST CANCER

VENKATESWARA RAO MUDUNURU

Department of Mathematics and Statistics, University of South Florida
Tampa, Florida 33620, USA

ABSTRACT. Artificial Neural Networks (ANNs), recently applied to a number of clinical, business, forecasting, time series prediction, and other applications, are computational systems consisting of artificial neurons called nodes arranged in different layers with interconnecting links. Among the available wide range of neural networks, most research is concentrated around feed forward neural networks called Multi-layer perceptrons (MLPs). One of the important components of an artificial neural network (ANN) is the activation function. This paper discusses properties of activation functions in multilayer neural network applied to breast cancer stage classification. There are a number of common activation functions in use with ANNs. The main objective in this work is to compare and analyze the performance of MLPs which has back-propagation algorithm using various activation functions for the neurons of hidden and output layers to evaluate their performance on the stage classification of breast cancer data.

1. INTRODUCTION

Artificial neural networks are important in the performance analysis of the breast cancer patient's survival analysis. Mathematically, they are a system of linked parallel equations which are solved simultaneously and iteratively. The power and usefulness of ANNs have been demonstrated in several applications including speech synthesis and recognition [1], diagnostic problems [2], medicine, business and finance, robotic control [3], signal processing, computer vision and many other problems. Neural Networks has a large appeal to many researchers due to their great closeness to the structure of the brain, a unique characteristic not shared by many traditional systems. An artificial neural network learns iteratively and weights are adjusted through training to minimize error between output and target value. The ANN achieves convergence when no further variations can be made. This learning model is most commonly applied by back propagation network (BPN), which is a supervised learning network. BPN provides, in particular, the input and output training samples from specific problems, and the network learns from the correspondence among the samples [4].

Received February 7, 2016

A typical neural network consists of these parts; processing units, weighted interconnection, and activation functions in hidden layer. In this paper we have implemented various combinations of the activation functions and their performance is estimated. We call this as a full model. The training and testing accuracy are obtained for all the combinations of activation functions using the same set of input variables. The most efficient activation function combination identified in this process is implemented in designing a reduced network model by eliminating the least contributing variables from the full model.

2. LITERATURE REVIEW

The linear combination of hidden unit activations characterizes the nonlinearity that is hard to capture by a linear function. Also studies which are concerned with survival do not possess data with similar time features. For example time of interest for some particular cases is so small which can be classified into 2 classes (i.e. event or no event) and subjected to analysis. Deaths considering the length of study of stay in an intensive care unit were included as good combination by Tu and Guerriere. [5] G. Cybenko (1989), K. Hormik et al. (1989) in their research articles [6, 7] discussed about using sigmoidal functions generating sigmoidal outputs as universal approximators. However E. J. Hartman, et al. (1990) and J. Park, et al. (1991) also termed Gaussian outputs also as universal approximators [8, 9]. Hartman and Keeler (1991) proposed a new activation function called Gaussian bars [10]. Pao (1989) in his book “Adaptive Pattern Recognition and Neural Networks” discussed about using a combination of various activation functions [11]. Simon Haykin and Leung (1993) were very successful with using radial transfer functions [12]. Dorffner (1994) using conic section function networks introduced new transformation functions that change smoothly from sigmoidal to Gaussian-like [13]. Girauld, et al. (1995) introduced simplified Gaussian functions called Lorentzian transfer functions which are widely used in many research works [14]. Predictive method for a risk of liver cancer for hepatitis B virus carriers is developed by Yang [15]. A variety of activation functions with their applications are discussed in detail here [16, 17].

3. DATA

The information that we have used in this present study is obtained from SEER database registry. This data source SEER [18, 19] (Surveillance Epidemiology and End Results), which is a unique, reliable and essential resource for investigating the different aspects of cancer. The SEER database combines patient-level information on cancer site, tumor pathology, stage, and cause of death [19]. In this work, we preprocessed the SEER data for breast cancer to remove redundancies and missing information. The resulting data set had 47,167 malignant tumor records. The variables utilized in this work include tumor size, treatment, age, number of primary tumors, and grade of the tumor, marital status, stage, duration and race of the women. In this study we used 33152 (70%) data for training, 14015 (30%) data for testing the trained network.

4. ACTIVATION FUNCTIONS

The crucial step in MLP neural network structure is generating the net inputs by using a scalar-to-scalar function which is known as the "activation function" or "threshold function" or "transfer function" [20]. These activation functions are used to limit the amplitude of the output of a neuron. The typical activation functions which are used to solve the non-linear problems are sigmoid, tangent, softmax, radial basis functions among others. These functions further process the output of the neuron after initial processing has taken place and are non-linear in nature by transforming the weighted sum of inputs to an output value and do the final mapping. In most cases these functions squash the amplitude range to a limited value probably the normalized value. Interestingly the outputs of these functions are further processed by running more number of iterations unless the network attains the desired convergence. In back propagation learning the functions implemented should have the characteristics like the continuous, differentiable, and monotonically non-decreasing and output should be bounded.

As mentioned earlier, ANNs are mostly used in modeling nonlinear data. Neural networks because of its nonlinear structure are used either to approximate a posteriori probabilities for clustering/classification or to approximate probability densities of the training data [21, 22]. Nonlinearity is introduced into an MLP network in the form of an activation function for the hidden units. The nonlinearity in the network is the reason why MLPs are so powerful. Below are few important papers surveyed which show that the choice of transfer functions is considered by some experts to be as important as the network architecture and learning algorithm.

Two most popular feed forward neural networks models, the multi-layer perceptron (MLP) and the Radial Basis Function (RBF) networks, are based on specific architectures and the transfer functions. Below are few activation functions in detail. Figure 1 gives the functional behavior of few such activation functions.

4.1. **Identity Function.**

The Identity function is also known a linear function. The output of the function is same as the input variable. Sometimes a constant is used to multiply it to form a linear function with scaled magnitude. The activation function needs to introduce nonlinearity into the networks for the network to be robust.

$$\begin{aligned} &f(x) = x \\ &f(x) = kx, \text{ where } k \text{ is a scaling constant} \end{aligned} \tag{4.1}$$

4.2. **Binary Step Function.** This function is also known as the Heaviside function or threshold function or hard limit function, with threshold theta. The output is always a binary value and it is decided by the function.

$$f(x) = \begin{cases} 0, & \text{if } x \leq \theta \\ 1, & \text{if } x < \theta \end{cases} \tag{4.2}$$

4.3. **Saturating linear function.** This function is also known as ramp function or piece wise linear sigmoid function [23] combines the Heaviside function with a linear output function.

$$f(x) = \begin{cases} 0, & x \leq 0 \\ x, & 0 < x < 1 \\ 1, & x \geq 1 \end{cases} \tag{4.3}$$

4.4. **Sigmoid Functions.** Sigmoidal output functions smooth out many shallow local minima in the total output functions of the network. For classification type of problems this may be desirable, but for general mappings it limits the precision of the adaptive system [24]. This is the most commonly used transfer function in MLP as it gives good results in most cases and can dramatically reduce the computation burden of training. The term sigmoid mean a graph which is 'S-shaped' curve. It is most commonly used function in the neural networks where the training is implemented by using the back propagation algorithms. The significance of this function is that the computation capacity for training is reduced and can be distinguished easily.

Uni-polar sigmoid

The output of this function is bounded to $[0, 1]$. The function gets zero to as the value of x tends to infinity in the negative side. Its analytic equation is given below.

$$f(x) = \frac{1}{1 + e^x} \tag{4.4}$$

Bi-Polar Sigmoid Function

The bi-polar sigmoid function is similar to the uni-polar sigmoid except that the limits of the output range between $[-1, 1]$.

$$f(x) = \frac{1 - e^x}{1 + e^x} \tag{4.5}$$

Bipolar binary and uni-polar binary are called as hard limiting activation functions used in discrete neuron model. Uni-polar continuous and bipolar continuous are called soft limiting activation functions are called sigmoidal characteristics.

4.5. **Hyperbolic Tangent Function.** The hyperbolic transfer function also ranges between $[-1, 1]$. This function is implemented in the replication of the sigmoid function where the output range is varying between -1 to 1.

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{\sinh x}{\cosh x} = \tanh x \tag{4.6}$$

4.6. **Radial basis functions (RBFs).** As MLP's implement sigmoidal transfer functions, RBFs typically use Gaussian functions. Both types of networks are universal approximators. This is an important, but almost trivial property, since any network using non-polynomial transfer functions are always universal approximators. The speed of convergence and the complexity of these networks to solve a given problem is more interesting.

$$g(x, c) = g(\|x - c\|) \tag{4.7a}$$

$$y(x) = \sum_{i=1}^{N} w_i g(\|x - c_i\|) \tag{4.7b}$$

where (4.7b) is represented as a sum of N radial basis functions and each of them are associated with a different center c_i and weighted by an appropriate weight and can be obtained by the matrix methods of linear least squares [25].

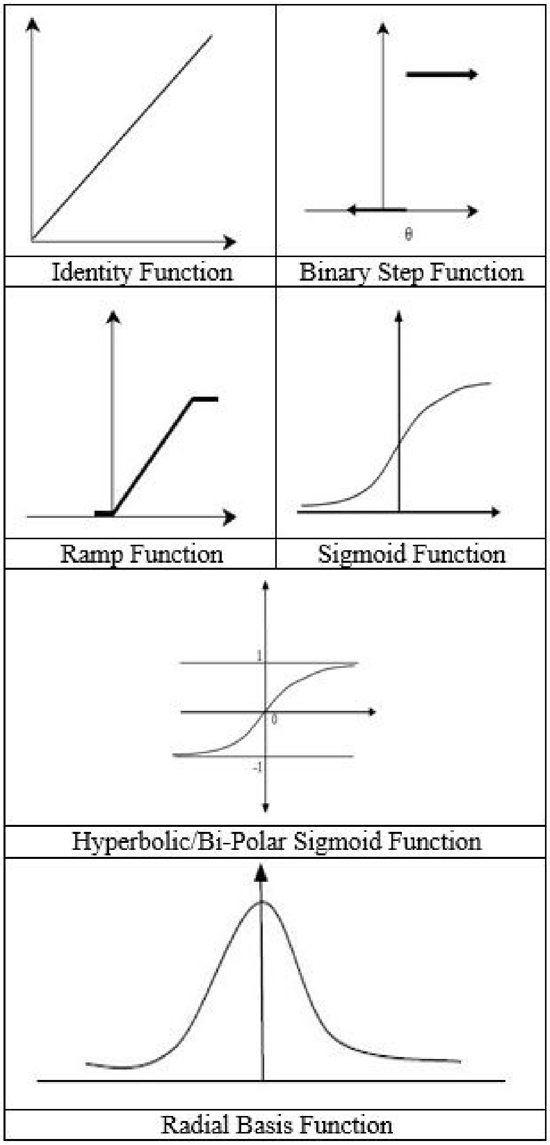
5. FRAME WORK FOR MODELING

The entire work in the paper is divided into two steps. The first step deals with the stage classification of breast cancer considering all the variables mentioned in the previous section as inputs and stage of cancer as the output variable. We consider these models as full models as we use all variables listed above as inputs. The second step involves identifying a reduced model using the best activation pair identified in the first step and by eliminating the input variables which have less than 5% normalized importance in performance of cancer stage classification.

6. METHODOLOGY

The most widely used supervised neural classifier called multilayer perceptron with one hidden layer is used in this study. Each input neuron is the input layer represents one of the input features such as age, tumor size, etc. while each neuron in the output layer corresponds to the stage (1 or 2 or 3 or 4) of cancer.

A multilayer perceptron (MLP) network is trained and tested using the inputs for the stage classification. It is proven earlier in many situations that MLPs possess the ability to learn and give the better performance especially in the case of classification. The complexity of the MLP network can be changed by varying the number of layers and the number of units in each layer. With a proper combination of hidden units



Identity Function

Binary Step Function

Ramp Function

Sigmoid Function

Hyperbolic/Bi-Polar Sigmoid Function

Radial Basis Function

Figure 1. Activation Functions

and activation functions, it has been shown that MLPs can approximate virtually any complex nonlinear function to any desired accuracy [25]. What kind of activation functions should be selected is very important and also is a very difficult problem. In MLP neural networks, when input values pass into nodes through interconnections, they are multiplied by the weight associated with these interconnections and summed. Then the activation function determines the output value of this node. The choice of activation functions will strongly influence the complexity and performance of ANNs. Theoretically, any differentiable function can be used as an activation function. However, an activation function having a nonlinear character is so important in order to be able to discriminate the complex relationships that exist in the feature space.

The MLP network has to be well trained before it is able to perform specific task such as prediction. In ANN, training outcome is determined by parameter setting of prediction errors and convergence speed. This consists of various parameter considerations like number of input neurons, number of hidden layers and hidden units, epochs or learning cycles, learning rate and coefficient of momentum. For all the models in this paper the learning rate is set to 0.4, and momentum to 0.9. In our work, we trained and tested six MLP networks using different combinations of activation functions for the hidden and output layers. The corresponding number of hidden nodes, percent of correct classification and ROC area are recorded and tabulated. The best activation function combination for stage classification is selected based on these results.

In the first part of work, the activation functions hyperbolic tangent and sigmoid functions in hidden layer are mapped to softmax, hyperbolic tangent, and sigmoid functions in output layer respectively. These combinations gave us six MLP networks. After identifying the best activation function pair that best represents the problem, in the second step, we worked further to find a reduced model ANN, if any, by eliminating the input variables which perform least in classifying the stages of breast cancer.

7. EVALUATION OF A MODEL PERFORMANCE

The predictions from each model were ranked by comparing number of correct classifications, positive predictive values (PPVs), overall accuracy and the comparison of area under ROC curves. The values of training and testing the full and reduced models were evaluated and tabulated. The model with minimum hidden units, minimum number of misclassifications, maximum ROC area is chosen as winner model.

8. RESULTS AND DISCUSSION

The main objective of this paper is to compare the performance of an MLP network by using different activation functions. For all the MLPs with different activation functions hidden nodes are selected automatically based on the requirement for training. The best number of hidden nodes required in the hidden layer depends on the number of input variables, amount of noise in the targets, activation function used.

In the Table 2 and Table 3, we have listed all the results of 6 full models with the percentage of correct predictions, PPVs, overall accuracy during training and testing the MLP network. The stage wise ROC area under curve values is given in Table 1. From these tables, the model with hyperbolic tangent and softmax function has a better prediction with less number of hidden nodes. Figure 2 gives the ROC of the selected model. Although the sigmoid-softmax pair has comparatively same results like hyperbolic tangent-softmax pair, we selected hyperbolic tangent-softmax pair for the following reasons. A MLP model with the best performance using less number of hidden units is considered as the best ANN representing the problem. Hyperbolic tangent - softmax model uses only 8 hidden units whereas softmax-sigmoid network uses 9 hidden units. Also since the hyperbolic tangent activation function has a derivative, it can be used with gradient descent based training methods. The hyperbolic tangent activation function is perhaps the most common activation function used for neural networks. The hyperbolic tangent function provides similar scaling to the sigmoid activation function, however, the hyperbolic tangent activation function has a range from -1 to 1. Because of this greater numeric range the hyperbolic activation function is chosen in place of the sigmoid activation function.

After selecting the combination of better performing activation functions, we attempted to find a reduced model by eliminating the input variables which have less than 5% normalized importance in performance of breast cancer stage classification. From Table 4 the input variables race, marital status and grade are the variables which are below 5% normalized importance for the selected full model and are eligible for elimination. Eliminating these input variables we modeled a reduced MLP neural network model to perform stage classification of breast cancer. This reduced model with hyperbolic tangent-softmax activation function performed equally with the full model, using only few input and hidden units.

Finally we compared the performance of selected full model with the reduced model by comparing the architecture of the network, overall performance based on accuracy, PPVs, and ROCs. The reduced model used only 8 input variables, 6 hidden units to classify cancer stage when compared with 22 inputs and 8 hidden units of full model. Table 5 has the training and testing classification results for the reduced model and Figure 2 gives the ROC of both full and reduced models. Table 6 has the comparison details of performance of full and reduced model. Reduced model performed almost close to the full model but with fewer units in input and hidden layers. Table 7 has the comparison of ROCs for the full and reduced model. Clearly the reduced model is performing better than the full model and hence this model is opted as winner model for breast cancer stage classification.

9. CONCLUSION

In this paper a MLP neural network for the breast cancer stage classification with given input parameters is considered. A neural network with a best activation function makes the diagnostic procedure, better, quicker and accurate. Though training a network may be time consuming, but once well trained, they show reliable results. The proposed method will be very handy to a practicing oncologist. They only have to go through the input pattern with the defined input variables and the network gives them the results about the classification of breast cancer stages. This reduces the cost of breast cancer stage classification and with proper computer programming we can make this facility accessible for a larger number of breast cancer patients.

Table 1. Stage wise area under ROC values of full models

Activation Functions	**AUROC Stages**			
Full Models\Stages	1	2	3	4
HT–Softmax	0.911	0.866	0.910	0.910
HT–HT	0.910	0.866	0.882	0.895
HT–Sigmoid	0.910	0.859	0.909	0.886
Sigmoid–Softmax	0.912	0.868	0.913	0.919
Sigmoid–HT	0.909	0.863	0.862	0.881
Sigmoid–Sigmoid	0.910	0.862	0.909	0.882

Table 2. ANN training results of full models

		Positive Predictive Probabilities				
Full Model details	**Number of hidden units**	**P(1|1)**	**P(2|2)**	**P(3|3)**	**P(4|4)**	**Overall Accuracy**
HT–SM	8	88.9%	75.0%	41.9%	33.8%	79.0%
HT–HT	9	91.7%	73.7%	45.3%	26.3%	79.8%
HT–S	9	90.4%	76.4%	0%	0%	77.0%
S–SM	9	89.5%	74.5%	46%	26%	79.1%
S–HT	9	90.6%	75.1%	40.7%	1.2%	79.0%
S–S	9	91.7%	73.7%	50.9%	0%	79.4%

HT-Hyperbolic Tangent, S-Sigmoid, SM-Softmax

Table 3. ANN testing results of full models

		Positive Predictive Probabilities				
Full Model details	**Number of hidden units**	**P(1|1)**	**P(2|2)**	**P(3|3)**	**P(4|4)**	**Overall Accuracy**
HT–SM	8	88.9%	75.0%	39.3%	28.8%	78.8%
HT–HT	9	92.1%	72.7%	43.0%	26.1%	79.5%
HT–S	9	90.9%	76.1%	0%	0%	77.6%
S–SM	9	89.8%	74.5%	45.1%	23.7%	79.1%
S–HT	9	90.8%	73.5%	43.1%	0.8%	78.5%
S–S	9	91.7%	72.5%	51.2%	0%	79.0%

HT-Hyperbolic Tangent, S-Sigmoid, SM-Softmax

Table 4. Input variables and their normalized importance

Inputs	**Imp**	**N. Imp**
M_status	.021	3.8%
Race	.018	3.4%
Grade	.025	4.6%
Treatment	.127	23.4%
Age	.059	10.9%
Numprims	.085	15.7%
Tumor_size	.543	100.0%
Duration	.121	22.3%

Imp - Importance, N. Imp- Normalized Importance

Table 5. Training and testing results of reduced model

		Positive Predictive Probabilities				
Reduced Model details	**ANN Architecture I–H–O**	**P(1|1)**	**P(2|2)**	**P(3|3)**	**P(4|4)**	**Overall Accuracy**
Training	8–6–4	89.8%	74.2%	49.8%	30.3%	79.5%
Testing	8–6–4	90.0%	73.5%	49.2%	24.9%	79.0%

Table 6. Comparison of full and reduced model classification

Details	**Full model**		**Reduced model**	
Architecture I–H–O	**22–8–4**		**8–6–4**	
PPVs	**Training**	**Testing**	**Training**	**Testings**
P(1|1)	88.9%	88.9%	88.9%	88.9%
P(2|2)	75.0%	75.0%	74.2%	73.5%
P(3|3)	41.9%	39.3%	49.8%	49.2%
P(4|4)	33.8%	28.8%	30.3%	24.9%
Overall accuracy	79.4%	79.0%	79.5%	79.0%

I-Input units; H-Hidden units; O- Output units

Table 7. ROC comparison for full and reduced models

Models	**Stage-1**	**Stage-2**	**Stage-3**	**Stage-4**
Reduced Model	0.911	0.868	0.912	0.915
Full Model	0.911	0.866	0.910	0.910

I-Input units; H-Hidden units; O-Output units

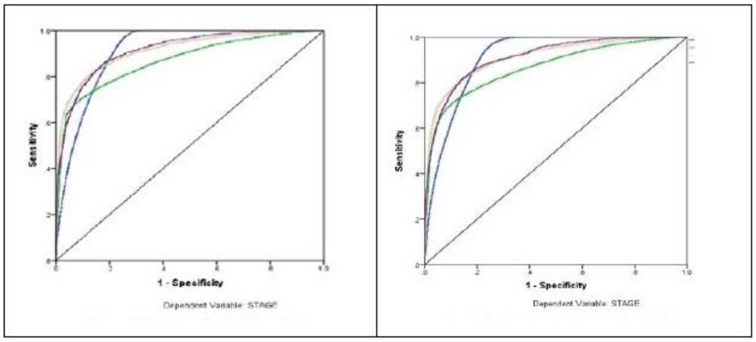


Figure 2. ROC of Full and Reduced Models

REFERENCES

[1] G. E. Dahl et al, *Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition*, IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, pp. 30–42, January 2012.

[2] Q. Kadhim, *Artificial Neural Networks in Medical Diagnosis*, IJCSI International Journal of Computer Sciences Issues, vol. 8, pp. 150–154, March 2011.

[3] Marsel Mano et al, *An Artificial Neural Network Based Robot Controller that Uses Rats Brain Signals*, Robotics, vol. 2, pp. 54–65, 2013.

[4] J. E. Dayhoff and J. M. De Leo, *Artificial neural networks Opening the Black Box*, Cancer, vol. 91, pp. 1615–1635, 2001.

[5] Jv Tu and M. R. Guerriere, *Use of a neural network as a predictive instrument for length of stay in the intensive care unit following cardiac surgery*, Proceedings of the Annual Symposium on Computer Application in Medical Care, pp. 666–672, 1992.

[6] G. Cybenko, *Approximation by Superpositions of a Sigmoidal Function*, Mathematics of Control, Signals, and Systems, vol. 2, pp. 303–314, 1989.

[7] M. Stinchcombe, H. White, and K. Hornik, *Multilayer feedforward networks are universal approximators*, Neural Networks, vol. 2, no. 5, pp. 359–366, 1989.

[8] J. D. Keeler and J. M. Kowalski and Eric J. Hartman, *Layered Neural Networks with Gaussian Hidden Units as Universal Approximations*, Neural Computation, vol. 2, no. 2, pp. 210–215, 1990.

[9] Sandberg J. Park et al, *Universal approximation using radial basis function networks*, Neural Computation, vol. 3, pp. 246–257, 1991.

[10] E. Hartman et al, *Predicting the future: Advantages of semilocal units*, Neural Computation, vol. 3, pp. 566–578, 1991.

[11] Y. H. Pao, *Adaptive pattern recognition and neural networks.*, in Addison-Wesley Longman Publishing Co., Inc., 1989.

[12] H. Haykin, *Rational function neural network, Neural Computation*, vol. 5, no. 6, pp. 928–938, 1993.

[13] G. Dorffner, *United framework for MLPs and RBFNs: Introducing conic section function networks*, Cybernetics and Systems: An International Journal, vol. 25, no. 4, pp. 511–554, 1994.

[14] B. G. Giraud et al, *Lorentzian neural nets, Neural Networks*, vol. 8, no. 5, pp. 757–767, 1995.

[15] H. I. Yang et al, *Nomograms for risk of hepatocellular carcinoma in patients with chronic hepatitis B virus infection*, Journal of Clinical Oncology, vol. 28, no. 14, pp. 2437–2444, 2010.

[16] W. Duch and N. Jankovski, *Survey of Neural Transfer Functions*, Neural Computing Surveys 2, vol. 2, pp. 163–212, 1999.

[17] S. Gomes et al, *Comparison of New Activation Functions in Neural Network for Forecasting Financial Time Series*, Neural Comput & Applic, vol. 20, pp. 417–439, 2011.

[18] P. Surveillance, *Surveillance, Epidemiology, and End Results (SEER) Program*, National Cancer Institute, DCCPS, Surveillance Research Program, May 2011.

[19] G. A. Colditz et al, *Risk factors for breast cancer according to estrogen and progesterone receptor status*, Journal of the National Cancer Institute, vol. 96, no. 3, pp. 218–228, 2004.

[20] Ruck D. W. and Rogers S. K. and Kabrisky M. and Oxley M. E. and Suter B. W., *The multilayer perceptron as an approximation to a Bayes optimal discriminant function. Neural Networks*, IEEE Transactions, vol. 1, no. 4, pp. 296–398, 1990.

[21] Nigrin A., *Neural Networks for Pattern Recognition*, MIT Press, 1993.

[22] Ripley B. D., *Pattern recognition via neural networks. A volume of Oxford Graduate Lectures on Neural Networks*, Oxford University Press, 1996.

[23] Duch W. and Jankowski N., *Transfer functions: hidden possibilities for better neural networks.*, In ESANN, pp. 81–94, 2001.

[24] Karlik B. and Olgac A. V., *Performance analysis of various activation functions in generalized MLP architectures of neural networks.*, International Journal of Artificial Intelligence and Expert Systems, vol. 1, no. 4, pp. 111–122, 2010.

[25] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall Inc, 1994.