

Research Article

A Clinical Decision Support System for Femoral Peripheral Arterial Disease Treatment

Alkın Yurtkuran,¹ Mustafa Tok,² and Erdal Emel¹

¹ Department of Industrial Engineering, Faculty of Engineering, Görükle Campus, Uludağ University, 16059 Bursa, Turkey

² Department of Thoracic and Cardiovascular Surgery, Faculty of Medicine, Görükle Campus, Uludağ University, 16059 Bursa, Turkey

Correspondence should be addressed to Alkın Yurtkuran; alkin@uludag.edu.tr

Received 30 July 2013; Revised 4 November 2013; Accepted 7 November 2013

Academic Editor: Gabriel Turinici

Copyright © 2013 Alkın Yurtkuran et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

One of the major challenges of providing reliable healthcare services is to diagnose and treat diseases in an accurate and timely manner. Recently, many researchers have successfully used artificial neural networks as a diagnostic assessment tool. In this study, the validation of such an assessment tool has been developed for treatment of the femoral peripheral arterial disease using a radial basis function neural network (RBFNN). A data set for training the RBFNN has been prepared by analyzing records of patients who had been treated by the thoracic and cardiovascular surgery clinic of a university hospital. The data set includes 186 patient records having 16 characteristic features associated with a binary treatment decision, namely, being a medical or a surgical one. K-means clustering algorithm has been used to determine the parameters of radial basis functions and the number of hidden nodes of the RBFNN is determined experimentally. For performance evaluation, the proposed RBFNN was compared to three different multilayer perceptron models having Pareto optimal hidden layer combinations using various performance indicators. Results of comparison indicate that the RBFNN can be used as an effective assessment tool for femoral peripheral arterial disease treatment.

1. Introduction

Various engineering techniques have been adapted to health care delivery systems and the quality of health care services has been improved using these artificial intelligence techniques. It has been proven that introducing machine learning tools into clinical decision support systems can easily increase the decision accuracy and decrease costs and the dependency on highly qualified specialists. Since artificial neural networks (ANN) can easily be trained for identifying the patterns and extracting rules using a small number of cases, they are widely used as a powerful tool for clinical decision support systems [1].

Peripheral arterial disease (PAD) is a common pathologic disease worldwide. Peripheral arterial disease is a disease in which plaque, which is made up of fat, cholesterol, calcium, fibrous tissue, and other substances in the blood, builds up in the arteries that carry blood to head, organs, and limbs. PAD affects more than 30 million people worldwide, and while it

can strike anyone, it is most common in people over age 65 [2].

PAD is associated with a significant burden in terms of morbidity and mortality, due to claudication, rest pain, ulcerations, and amputations. In case of mild or moderate peripheral arterial diseases, a medical or conservative therapy can be chosen but the gold-standard treatment of severe PAD is a surgical or an endovascular revascularization [2]. However, up to 30% of patients are not candidates for such interventions, due to excessive surgical risks or unfavorable vascular involvements. The presence of diffuses and multiple and distal arterial stenosis renders successful revascularization sometimes impossible. These “no-option” patients are left to medical therapy, which may slow the progression of disease at best [3].

It is very difficult to decide whether surgical or medical treatment is the best option since PAD depends on many factors like anatomic location, symptoms, comorbidities, and risk about cardiac condition or anesthesia. Cardiovascular

surgeons should prefer the best appropriate choice of treatment and most of the time the decision allows the surgeon with his own experience. Cardiovascular specialists widely use intersociety Consensus for the classification of PADs' (TASC II) (Trans-Atlantic intersociety Consensus), which is based on the anatomic locations of lesions [3].

In this work, we present a clinical treatment decision support system using a radial basis function neural network (RBFNN) in order to help doctors to make an accurate treatment decision for patients having femoral PAD. Proposed RBFNN was compared to three different multilayer perceptron (MLP) networks and results indicate that the proposed RBFNN outperforms MLP networks. Based on our extensive literature review, no previous study was carried out which included a decision support system for clinical treatment of femoral PAD.

The remainder of this paper is organized as follows. Section 2 summarizes previous studies; Section 3 covers the clinical data and input and output features of the proposed model. Section 4 gives a brief introduction to the RBFNN and experiments. Related results are given in Section 5 and finally Section 6 concludes the paper.

2. Related Work

In recent years, there have been many studies that focused on decision support systems to improve the accuracy of decisions for diagnosis and treatment of diseases. Such decision support systems frequently depend on ANN-based perceptive algorithms that are built upon previous patient records.

To cite a few but significant works of others, Mehrabi et al. [4] used a MLP network and a RBFNN to classify chronic obstructive pulmonary (COPD) and congestive heart failure (CHF) diseases. They used Bayesian regularization to enhance the performance of MLP network. Moreover, they integrated K-means clustering algorithm and k-nearest neighborhood, to define centers for hidden neurons and to identify the spread, respectively. They have shown that both COPD and CHF have been classified using the MLP networks and the RBFNN accurately.

Subashini et al. [5] proposed a polynomial kernel for the support vector machine (SVM) and the RBFNN for ascertaining the diagnostic accuracy of cytological data obtained from the Wisconsin breast cancer database. They have shown that RBFNN outperformed SVM for accurately classifying the tumors. Lewenstein [6] used RBFNN as a tool for diagnosis of coronary artery disease. The research was performed using 776 data records and over 90% accuracy was achieved for classifying.

A short review of recent studies reveal numerous use of ANN techniques for diagnosis of diabetes mellitus [7–12], chest diseases [13–17], Parkinson disease [18, 19], breast cancer [5, 20–23], thyroid disease [24–26] and cardiovascular diseases [4, 6, 27–36].

Broomhead and Lowe [37] were the first to use the RBFNN in designing neural networks. In recent years, the RBFNN have attracted extensive research interest. [38–42]

Wu et al. [19] used RBFNN to accurately identify Parkinson's disease. The data for training the RBFNN was obtained by means of deep brain electrodes implanted into a Parkinson's disease patient's brain. The output of the study indicated that RBFNNs could be successfully designed and used to identify tremors on set pattern even for small number of spikes.

3. The Clinical Data

The input data set for training ANNs has been obtained from discharge reports dated from 2008 to 2012 within patient records of the department of thoracic and cardiovascular surgery clinic of a university hospital. 186 records with 114 male patients aged around 53 ± 7 and with 72 female patients aged as 58 ± 5 have been analyzed. Each patient's report contains one final treatment decision, which is taken here as an output class value of the corresponding input data set that is as follows.

- (i) Class 1: medical treatment decision (89 patients).
- (ii) Class 2: surgery or endovascular treatment decision (97 patients).

All samples have a total of 16 features and these features were determined by consultations with cardiologists, surgeons, and anesthesiologists. Features, output classes and their normalized values are given in Table 1. Description of selected features is summarized in Tables 2–5.

4. Radial Basis Function Neural Network (RBFNN)

The RBFNN [43] has a feed forward architecture with 3 layers: (i) an input layer, (ii) a hidden layer, and (iii) an output layer. A typical RBFNN is shown in Figure 1. The input layer of m nodes accepts m -dimensional features as input data vector. The hidden layer, which is fully connected to the input layer, is composed of n radial basis function neurons. Each hidden layer neuron operates as a radial basis function that does a nonlinear mapping of feature space into output space. The output layer consists of c neurons, which calculate the weighted sum of the output of the each hidden layer node.

The most commonly employed radial basis function for hidden layers is the Gaussian function [44, 45] and is determined by mean vectors (cluster centers) μ_j and covariance matrices C_j where $j = 1, \dots, n$. Covariance matrices are assumed to be in the form $C_j = \sigma_j^2 \mathbf{I}$.

Let $\Phi_j(\mathbf{x})$ be the Gaussian function representing the j th hidden node defined as

$$\Phi_j(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \mu_j\|^2}{2\sigma_j^2}\right), \quad (1)$$

where $\mathbf{x} = [x_1, x_2, \dots, x_m]^T$ is the input feature vector, $\mu_j = [\mu_{1j}, \mu_{2j}, \dots, \mu_{mj}]^T$ and σ_j^2 are the mean vector and the

TABLE 1: Features and their normalized values.

Feature	Comment
Age (years)	Divided by 100
Sex	Female = 0, male = 1
Fontaine stage	Stage I = 0, stage II-a = 0, stage II-b = 2, stage III = 3, stage IV = 4 (see Table 4)
Lesion type (TASC classification)	Type A = 0, type B = 1, type C = 3, type D = 4 (see Table 5)
Sensitivity to anesthesia	Low = 0, medium-high = 1
Distal bed	Absence = 0, presence = 1
Embolism (percent)	Divided by 100
LDL cholesterol level	Normal = 0, near/above normal = 1, BH = 2, high = 3, very high = 4 (see Table 3)
Smoking	Absence = 0, presence = 1
Exsmoker	Absence = 0, presence = 1
Hypertension	Absence = 0, presence = 1
Blood pressure	Normal = 0, pre-HTN = 1, stage I = 2, stage II = 3 (see Table 2)
Diabetes mellitus	Absence = 0, presence = 1
Other peripheral disease history	Absence = 0, presence = 1
Family history	Absence = 0, presence = 1
Current medical treatment	Absence = 0, presence = 1
Treatment decision	Medical treatment = -1, operation = 1

TABLE 2: Blood pressure level categories in adults.

Classification	Systolic pressure (mm Hg)	Diastolic pressure (mm Hg)
Normal	<120	<80
Prehypertension	120–139	80–89
Stage I	140–159	90–99
Stage II	>160	>100

TABLE 3: Cholesterol level categories in adults.

LDL cholesterol level (mg/dL)	LDL cholesterol category
<100	Optimal
100–129	Near optimal/above optimal
130–159	Borderline high
160–189	High
>190	Very high

TABLE 4: Fontaine stages [2].

Stages	Details
Stage I	Asymptomatic, incomplete blood vessel obstruction
Stage II-a	Claudication at a distance of greater than 200 meters
Stage II-b	Claudication distance of less than 200 meters
Stage III	Rest pain, mostly in the feet
Stage IV	Necrosis and/or gangrene of the limb

variance of the j th neuron, respectively. The k th output of the RBFNN is computed according to (2)

$$y_k = \sum_{j=1}^n w_{jk} \Phi_j(\mathbf{x}) + w_{0k}. \quad (2)$$

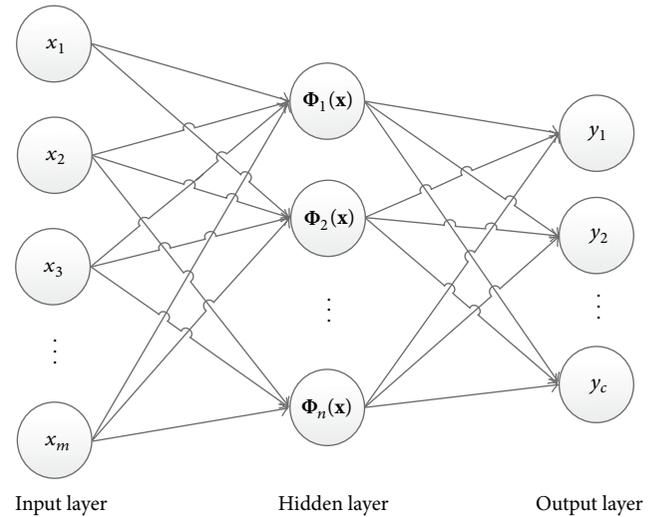


FIGURE 1: An example of RBFNN.

In (2), $\mathbf{w}_k = [w_{1k}, w_{2k}, \dots, w_{nk}]$ is the vector of the weights between hidden and output layer and w_{0k} is the bias for $k = 1, \dots, c$. In order to design a RBFNN, the value of mean vectors (μ_j) representing the location of cluster centers and variances (σ_j^2) for hidden neurons have to be calculated first. K-means clustering algorithm is used to determine the value of mean vectors which is given as follows.

Step 1. Initialize by choosing m random values for n hidden nodes ($\mu_{ij}, i = 1, \dots, m, j = 1, \dots, n$) as initial cluster centers.

Step 2. Assign a randomly selected input data sample \mathbf{x} to the nearest j th cluster center using the Euclidean norm.

Step 3. Recalculate μ_j including the assigned sample.

TABLE 5: TASC classification [3].

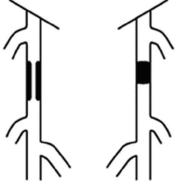
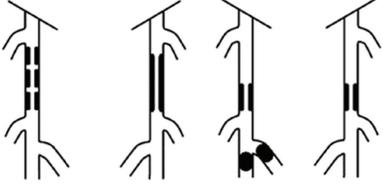
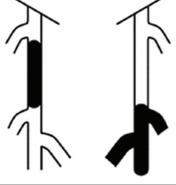
Lesion type	Description	Visual display
Type A	(i) Single stenosis ≤ 10 cm in length (ii) Single occlusion ≤ 5 cm in length	
Type B	(i) Multiple lesions, each ≤ 5 cm (ii) Single stenosis or occlusion ≤ 15 cm (iii) Single or multiple lesions in the absence of tibial vessels (iv) Heavily calcified occlusion ≤ 5 cm (v) Single popliteal stenosis	
Type C	(i) Multiple stenosis or occlusions totaling ≥ 15 cm (ii) Recurrent stenosis or occlusions that need treatment	
Type D	(i) Chronic total occlusions of common femoral artery or superficial femoral artery (ii) Chronic total occlusion of popliteal artery	

TABLE 6: Confusion matrix for binary classification.

Class/classified	As positive	As negative
Positive	tp	fn
Negative	fp	tn

Step 4. Repeat Steps 2 and 3 until mean vectors do not change ($\mu_j^{\text{new}} \cong \mu_j^{\text{old}}$).

The number of hidden neurons n , which should be determined experimentally, is effective on the performance of the RBFNN. Generally, it is assumed that variances of all clusters are identical and equal to σ^2 which is calculated as follows

$$\sigma^2 = \frac{\eta d^2}{2}, \quad (3)$$

where d is the maximum distance between cluster centers and η is an empirical scale factor and controls the smoothness of the nonlinear mapping function. Once the location of centers and their variances are determined, weights between the hidden layer and the output layer can be calculated. Equation (2) may be rewritten in the vector form as

$$\mathbf{Y} = \mathbf{H} \cdot \mathbf{W}. \quad (4)$$

In (4), \mathbf{Y} is the $(n \times 1)$ dimensional output vector, \mathbf{H} is the $(n \times (m + 1))$ dimensional hidden neuron matrix, and \mathbf{W} is the $((m + 1) \times 1)$ dimensional weight vector. To reduce the computational effort \mathbf{W} is directly calculated from the least squares pseudoinverse by (5)

$$\mathbf{W} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{Y}. \quad (5)$$

5. Experiments

5.1. Measures for Performance Evaluation. In our experiments, in order to evaluate the performance of the proposed RBFNN effectively and accurately, several performance indicators such as area under the receiving operating characteristics curve (AUC), *accuracy*, *sensitivity (recall)*, *specificity*, *positive predictive value (PPV) (precision)*, *negative predictive value (NPV)*, *F-score*, and *Yuden Index* are analyzed [35, 36]. All these performance indicators are determined by using a confusion matrix, which is composed of the results of a binary (true/false) classification in terms of true positive (tp), false positive (fp), false negative (fn), and true negative (tn) counts. A confusion matrix for a binary classification is presented in Table 6. *Accuracy* is used to assess the overall effectiveness of the classifier (see (6)). *Sensitivity* is the ratio of correctly classified samples to all samples in that class (see (7)). *Specificity* measures the proportion of negatives, which are correctly identified (see (8)). PPV is the accuracy

TABLE 7: Selected MLP networks.

Network name	Training algorithm	Hidden activation function	Output activation function	Number of hidden units
MLP-13	BFGS	tanh	Logistic	13
MLP-23	BFGS	Identity	Logistic	23
MLP-7	CGA	Logistic	Identity	7

TABLE 8: Mean of performance indicators for MLP networks and RBFNN.

	MLP-13	MLP-23	MLP-7	RBFNN
AUC	0.873	0.839	0.793	0.949
Cutoff point	0.443	0.542	0.392	0.510
Accuracy	0.881	0.838	0.800	0.950
Sensitivity	0.896	0.835	0.816	0.953
Specificity	0.868	0.840	0.788	0.948
PPV	0.849	0.824	0.753	0.942
NPV	0.909	0.851	0.843	0.958
<i>F</i> -score	0.872	0.829	0.783	0.947
Yuden index	0.764	0.675	0.604	0.901
<i>H-L</i>	10.386	10.211	11.632	7.880

in a specified class (see (9)) and NPV is the proportion of cases with negative results that are correctly classified (see (10)). Finally, *F-measure* and *Yuden Index*, which are widely used performance indicators to assess neural network classification performances, are depicted in (11) and (12). Another important performance indicator of neural networks is the area under the receiving operating characteristics curve (AUC). Receiving operating characteristics curve is constructed by plotting the *sensitivity* versus (*1-specificity*) values for variety of cutoff points between 0.00 and 1.00. Furthermore, the *Hosmer-Lemeshow (H-L)* chi-square statistic is used as a numerical indicator of overall calibration

$$\text{Accuracy} = \frac{\text{tp} + \text{tn}}{\text{tp} + \text{fp} + \text{fn} + \text{tn}}, \quad (6)$$

$$\text{Sensitivity} = \frac{\text{tp}}{\text{tp} + \text{fn}}, \quad (7)$$

$$\text{Specificity} = \frac{\text{tn}}{\text{tn} + \text{fp}}, \quad (8)$$

$$\text{PPV} = \frac{\text{tp}}{\text{fp} + \text{tp}}, \quad (9)$$

$$\text{NPV} = \frac{\text{tn}}{\text{tn} + \text{fn}}, \quad (10)$$

$$F_{\text{score}} = \frac{2 \times \text{Sensitivity} \times \text{PPV}}{\text{Sensitivity} + \text{PPV}}, \quad (11)$$

$$\text{Yuden Index} = \text{sensitivity} + \text{specificity} - 1. \quad (12)$$

5.2. Computational Results. Neural networks are prone to overfitting, especially when there are only a limited number of data. In order to estimate the performance of the neural

networks accurately by reducing the bias and the variance on predicted results, 10-fold cross-validation method is used in this study. Multifold cross-validation, in which dynamic sets of validation and test data are used, is an efficient technique to avoid overfitting compared to regularization, early stopping, or data pruning especially when data are very scarce [43]. In 10-fold cross-validation, a data set is randomly partitioned into 10 equal subsamples having approximately equal number of samples from each class. Using this data set, while the RBFNN training is done by the first nine subsamples, the validation is done only by the last subsample. This training and testing process is repeated for 10 times by rotating each subsample to be used only once as the validation subsample. The mean and standard deviation of performance indicators for each neural network model are then reported.

In this study, as mentioned in Section 4, the cluster center locations for all Gaussian functions, which are employed as radial basis functions, are determined using K-means clustering algorithm. Network weights of the output layer are determined by the pseudoinverse method (4). Following preliminary tests, the empirical scale factor is set to $\eta = 0.6$. For simplicity and ease of calculation, it is assumed that all variances are identical and equal to σ^2 . A program is written in C++ language to employ the proposed RBFNN model.

The optimum number of hidden nodes for a RBFNN model should be carefully determined as it directly affects the performance of the network. In this study, in order to choose the optimum number of centers for the proposed network, several preliminary experiments are conducted by stepwise change of the number of centers from 2 to 50. For each case, an average mean square error (MSE) is calculated using the 10-fold cross-validation. Figure 2 shows the MSE values with respect to the number of centers. Referring to Figure 2, the minimum MSE = 0.036 is achieved for 29 clusters and therefore the number of hidden nodes was set to 29.

After attaining the optimal RBFNN, the performance is compared to three different Pareto optimal three-layer MLP networks. In our study, MLP models were generated and implemented using the ANN module provided within the STATISTICA software (v 11.0) published by the Statsoft, Inc. MLP networks were constructed using the Automated Network Search (ANS) strategy for creating predictive models of STATISTICA. Best three MLP networks were retained by the ANS, trying different number of hidden units (1–30), different input/output activation functions (identity, logistic, tanh, and exponential) and different training algorithms such as the Gradient Descent, the Broyden-Fletcher-Goldfarb-Shanno (BFGS) (Quasi-Newton), the Conjugate Gradient Algorithm (CGA), or the Levenberg-Marquardt Algorithm

TABLE 9: Comparison of MLP-13 and RBFNN.

	MLP-13		RBFNN		Statistical significance
	Mean \pm SD	95% CI	Mean \pm SD	95% CI	
AUC	0.873 \pm 0.018	0.862–0.885	0.949 \pm 0.028	0.931–0.966	+
Cutoff	0.443 \pm 0.010	0.437–0.449	0.510 \pm 0.011	0.503–0.517	+
Accuracy	0.881 \pm 0.016	0.871–0.891	0.950 \pm 0.022	0.936–0.964	+
Sensitivity	0.896 \pm 0.021	0.883–0.909	0.953 \pm 0.015	0.944–0.963	+
Specificity	0.868 \pm 0.018	0.857–0.879	0.948 \pm 0.030	0.929–0.966	+
PPV	0.849 \pm 0.023	0.835–0.864	0.942 \pm 0.034	0.920–0.963	+
NPV	0.909 \pm 0.019	0.897–0.921	0.958 \pm 0.013	0.949–0.966	+
<i>F</i> -score	0.872 \pm 0.018	0.861–0.883	0.947 \pm 0.024	0.932–0.962	+
Yuden index	0.764 \pm 0.033	0.744–0.785	0.901 \pm 0.044	0.873–0.928	+
<i>H-L</i>	10.386 \pm 2.125	9.069–11.703	7.880 \pm 1.557	6.915–8.845	+

TABLE 10: Comparison of MLP-23 and RBFNN.

	MLP-23		RBFNN		Statistical significance
	Mean \pm SD	95% CI	Mean \pm SD	95% CI	
AUC	0.839 \pm 0.018	0.828–0.850	0.949 \pm 0.028	0.931–0.966	+
Cutoff	0.542 \pm 0.016	0.532–0.552	0.510 \pm 0.011	0.503–0.517	+
Accuracy	0.838 \pm 0.017	0.827–0.848	0.950 \pm 0.022	0.936–0.964	+
Sensitivity	0.835 \pm 0.020	0.823–0.847	0.953 \pm 0.015	0.944–0.963	+
Specificity	0.840 \pm 0.018	0.829–0.851	0.948 \pm 0.030	0.929–0.966	+
PPV	0.824 \pm 0.021	0.811–0.836	0.942 \pm 0.034	0.920–0.963	+
NPV	0.851 \pm 0.019	0.839–0.862	0.958 \pm 0.013	0.949–0.966	+
<i>F</i> -score	0.829 \pm 0.018	0.818–0.840	0.947 \pm 0.024	0.932–0.962	+
Yuden index	0.675 \pm 0.034	0.654–0.697	0.901 \pm 0.044	0.873–0.928	+
<i>H-L</i>	10.211 \pm 3.409	8.098–12.324	7.880 \pm 1.557	6.915–8.845	–

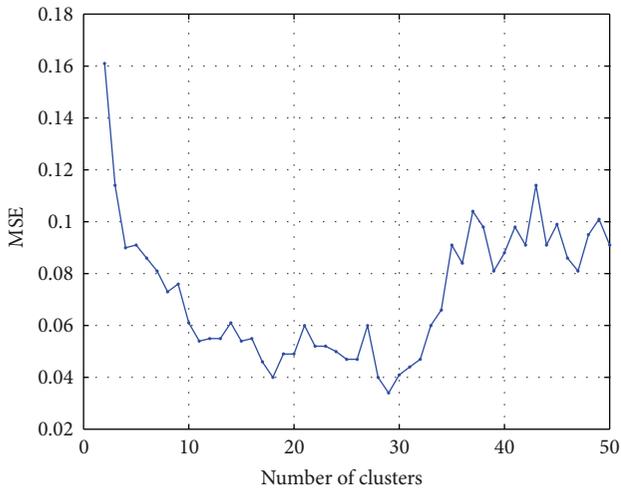


FIGURE 2: MSE versus number of clusters for proposed RBFNN.

using an error function of sum of squares. Moreover, a 10-fold cross-validation technique is selected to avoid overfitting and oscillation. The best three MLP networks which were determined using the ANS are summarized in Table 7. MLP-13 and MLP-23 employs the BFGS algorithm where the weights and biases are updated using the Hessian matrix

performance index at the current values of the weights and biases. BFGS has high memory requirements due to storing the Hessian matrix. On the other hand, MLP-7 utilizes the CGA, which is a fast training algorithm for MLP networks that proceeds by a series of line searches through error space. In CGA, learning rate and momentum are calculated adaptively in each iteration. In the ANS module, the learning rate is calculated by the Golden Search rule while the Fletcher and Reeves formula [46] is used for momentum calculations.

Table 8 lists the mean of performance indicator results using the 10-fold cross-validation method for each network. Considering Table 8, it is noticeable that the mean classification *accuracy* of RBFNN (0.950) is better than any one of MLP networks (MLP-13 = 0.881, MLP-23 = 0.838, and MLP-7 = 0.800). Prediction capabilities based on AUC show that the proposed RBFNN outperforms all other MLP networks (RBFNN = 0.949, MLP-13 = 0.873, MLP-23 = 0.839, and MLP-7 = 0.793). The average *sensitivity* values for MLP networks are 0.896, 0.835, and 0.816 for MLP-13, MLP-23, and MLP-7, respectively. On the other hand, proposed RBFNN gives an average *sensitivity* of 0.953, which indicates that the RBFNN performs better on classifying cases having positive condition. Based on *specificity*, the RBFNN (94.8%) is superior to MLP-13 (86.8%), MLP-23 (84.0%), and MLP-7 (78.8%). *F-measure* and *Yuden Index* are the most widely used stand-alone performance indicators for classification studies.

TABLE II: Comparison of MLP-7 and RBFNN.

	MLP-7		RBFNN		Statistical significance
	Mean \pm SD	95% CI	Mean \pm SD	95% CI	
AUC	0.789 \pm 0.019	0.778–0.801	0.949 \pm 0.028	0.931–0.966	+
Cutoff	0.392 \pm 0.009	0.386–0.398	0.510 \pm 0.011	0.503–0.517	+
Accuracy	0.800 \pm 0.020	0.787–0.812	0.950 \pm 0.022	0.936–0.964	+
Sensitivity	0.823 \pm 0.028	0.805–0.840	0.953 \pm 0.015	0.944–0.963	+
Specificity	0.782 \pm 0.017	0.772–0.792	0.948 \pm 0.030	0.929–0.966	+
PPV	0.746 \pm 0.021	0.733–0.759	0.942 \pm 0.034	0.920–0.963	+
NPV	0.850 \pm 0.025	0.834–0.865	0.958 \pm 0.013	0.949–0.966	+
F-score	0.782 \pm 0.022	0.769–0.796	0.947 \pm 0.024	0.932–0.962	+
Yuden index	0.605 \pm 0.041	0.579–0.630	0.901 \pm 0.044	0.873–0.928	+
H-L	11.632 \pm 2.169	10.288–12.976	7.880 \pm 1.557	6.915–8.845	+

F-measure and *Yuden Index* values are 0.947 and 0.901 for the proposed RBFNN while 0.872 and 0.764 for MLP-13, 0.829, and 0.675 for MLP-23 and 0.783 and 0.604 for MLP-7, respectively. The mean PPV's are 0.849, 0.824, 0.753 and 0.942, while the mean NPV's are 0.909, 0.851, 0.843, and 0.958 for MLP-13, MLP-23, MLP-7, and RBFNN, respectively. These findings also show that a RBFNN performs better than MLP networks. In general, all models were good-fit models based on the *H-L statistics* ($H-L < 12.0$).

In order to make precise and pairwise comparison between networks, two-tailed *t* tests are employed to show the statistical significance level of the difference of the mean of performance indicators for the RBFNN and MLP networks. Tables 9, 10, and 11 show the results of statistical tests. The mean, the standard deviation (SD), and the 95% confidence interval (CI) of each result are given in Tables 9–11. In the last column of Tables 9–11, a “+” sign denotes that the difference of performance indicator means is statistically significant at a 0.05 level, while a “–” sign indicates a difference which is not significant. The *t* test results clearly indicate that the difference between the proposed RBFNN network and MLP networks are statistically significant for all the indicators except the *H-L* statistic between MLP-23 and RBFNN. Therefore, it is evident that the proposed RBFNN is a better classifier for identifying the treatment type of femoral PAD's when compared to MLP networks.

6. Conclusion

In this work, an artificial intelligence model that determines the treatment type for femoral PAD is presented. The proposed model, which is based on the RBFNN framework, is compared to three Pareto optimal MLP networks using a repeated 10-fold cross-validation method for the reliability of results. The proposed RBFNN possesses superior performance than MLP networks in terms of performance measures such as AUC, *accuracy*, *sensitivity*, *specificity*, *positive predictive value*, *negative predictive value*, *F-score*, and *Yuden Index*. This work clearly indicates that RBFNN is a viable and powerful tool as a clinical decision support system for classifying the treatment options regarding femoral PADs. Future studies may cover using metaheuristic algorithms to

determine optimal design parameters of RBFNNs such as the number and the location of centers or variances of clusters and as a result enhance the classification performance.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] M. Frize, C. M. Ennett, M. Stevenson, and H. C. E. Trigg, “Clinical decision support systems for intensive care units: using artificial neural networks,” *Medical Engineering and Physics*, vol. 23, no. 3, pp. 217–225, 2001.
- [2] R. Fontaine, M. Kim, and R. Kieny, “Surgical treatment of peripheral circulation disorders,” *Helvetica Chirurgica Acta*, vol. 21, no. 5-6, pp. 499–533, 1954.
- [3] L. Norgren, W. R. Hiatt, J. A. Dormandy, M. R. Nehler, K. A. Harris, and F. G. R. Fowkes, “Inter-Society consensus for the management of peripheral arterial disease (TASC II),” *European Journal of Vascular and Endovascular Surgery*, vol. 33, no. 1, pp. S1–S75, 2007.
- [4] S. Mehrabi, M. Maghsoudloo, H. Arabalibeik, R. Noormand, and Y. Nozari, “Application of multilayer perceptron and radial basis function neural networks in differentiating between chronic obstructive pulmonary and congestive heart failure diseases,” *Expert Systems with Applications*, vol. 36, no. 3, pp. 6956–6959, 2009.
- [5] T. S. Subashini, V. Ramalingam, and S. Palanivel, “Breast mass classification based on cytological patterns using RBFNN and SVM,” *Expert Systems with Applications*, vol. 36, no. 3, pp. 5284–5290, 2009.
- [6] K. Lewenstein, “Radial basis function neural network approach for the diagnosis of coronary artery disease based on the standard electrocardiogram exercise test,” *Medical and Biological Engineering and Computing*, vol. 39, no. 3, pp. 362–367, 2001.
- [7] K. Polat and S. Güneş, “An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease,” *Digital Signal Processing*, vol. 17, no. 4, pp. 702–710, 2007.
- [8] K. Polat, S. Güneş, and A. Arslan, “A cascade learning system for classification of diabetes disease: generalized discriminant

- analysis and least square support vector machine,” *Expert Systems with Applications*, vol. 34, no. 1, pp. 482–487, 2008.
- [9] U. Ergün, N. Barýpçý, A. T. Ozan et al., “Classification of MCA stenosis in diabetes by MLP and RBF neural network,” *Journal of Medical Systems*, vol. 28, no. 5, pp. 475–487, 2004.
- [10] H. Temurtas, N. Yumusak, and F. Temurtas, “A comparative study on diabetes disease diagnosis using neural networks,” *Expert Systems with Applications*, vol. 36, no. 4, pp. 8610–8615, 2009.
- [11] V. Tresp, T. Briegel, and J. Moody, “Neural-network models for the blood glucose metabolism of a diabetic,” *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1204–1213, 1999.
- [12] O. Karan, C. Bayraktar, H. Gümüşkaya, and B. Karlik, “Diagnosing diabetes using neural networks on small mobile devices,” *Expert Systems with Applications*, vol. 39, no. 1, pp. 54–60, 2012.
- [13] G. Coppini, M. Miniati, M. Paterni, S. Monti, and E. M. Ferdeghini, “Computer-aided diagnosis of emphysema in COPD patients: neural-network-based analysis of lung shape in digital chest radiographs,” *Medical Engineering and Physics*, vol. 29, no. 1, pp. 76–86, 2007.
- [14] O. Er and F. Temurtas, “A study on chronic obstructive pulmonary disease diagnosis using multilayer neural networks,” *Journal of Medical Systems*, vol. 32, no. 5, pp. 429–432, 2008.
- [15] O. Er, C. Sertkaya, F. Temurtas, and A. C. Tanrikulu, “A comparative study on chronic obstructive pulmonary and pneumonia diseases diagnosis using neural networks and artificial immune system,” *Journal of Medical Systems*, vol. 33, no. 6, pp. 485–492, 2009.
- [16] E. Orhan, F. Temurtas, and A. Ç. Tanrikulu, “Tuberculosis disease diagnosis using artificial neural networks,” *Journal of Medical Systems*, vol. 34, no. 3, pp. 299–302, 2010.
- [17] P. S. Heckerling, B. S. Gerber, T. G. Tape, and R. S. Wigton, “Use of genetic algorithms for neural networks to predict community-acquired pneumonia,” *Artificial Intelligence in Medicine*, vol. 30, no. 1, pp. 71–84, 2004.
- [18] S. Pan, K. Warwick, J. Stein et al., “Prediction of Parkinson’s disease tremor onset using artificial neural networks,” in *Proceedings of the 5th IASTED International Conference on Biomedical Engineering (BioMED ’07)*, pp. 341–345, Acta Press Anaheim, Innsbruck, Austria, February, 2007.
- [19] D. Wu, K. Warwick, Z. Ma, J. G. Burgess, S. Pan, and T. Z. Aziz, “Prediction of Parkinson’s disease tremor onset using radial basis function neural networks,” *Expert Systems with Applications*, vol. 37, no. 4, pp. 2923–2928, 2010.
- [20] R.-F. Chang, W.-J. Wu, W. K. Moon, Y.-H. Chou, and D.-R. Chen, “Support vector machines for diagnosis of breast tumors on US images,” *Academic Radiology*, vol. 10, no. 2, pp. 189–197, 2003.
- [21] I. Christoyianni, E. Dermatas, and G. Kokkinakis, “Fast detection of masses in computer-aided mammography,” *IEEE Signal Processing Magazine*, vol. 17, no. 1, pp. 54–64, 2000.
- [22] D. Furundzic, M. Djordjevic, and A. J. Bekic, “Neural networks approach to early breast cancer detection,” *Journal of Systems Architecture*, vol. 44, no. 8, pp. 617–633, 1998.
- [23] M. N. Gurcan, H.-P. Chan, B. Sahiner, L. Hadjiiski, N. Petrick, and M. A. Helvie, “Optimal neural network architecture selection: improvement in computerized detection of microcalcifications,” *Academic Radiology*, vol. 9, no. 4, pp. 420–429, 2002.
- [24] K. Hoshi, J. Kawakami, M. Kumagai et al., “An analysis of thyroid function diagnosis using bayesian-type and SOM-type neural networks,” *Chemical and Pharmaceutical Bulletin*, vol. 53, no. 12, pp. 1570–1574, 2005.
- [25] L. Ozyilmaz and T. Yildirim, “Diagnosis of thyroid disease using artificial neural network methods,” in *Proceedings of the 9th International Conference on Neural Information Processing (ICONIP ’02)*, vol. 4, pp. 2033–2036, IEEE, Singapore, November, 2002.
- [26] F. Temurtas, “A comparative study on thyroid disease diagnosis using neural networks,” *Expert Systems with Applications*, vol. 36, no. 1, pp. 944–949, 2009.
- [27] W. G. Baxt, F. S. Shofer, F. D. Sites, and J. E. Hollander, “A neural computational aid to the diagnosis of acute myocardial infarction,” *Annals of Emergency Medicine*, vol. 39, no. 4, pp. 366–373, 2002.
- [28] K. M. Eggers, J. Ellenius, M. Dellborg et al., “Artificial neural network algorithms for early diagnosis of acute myocardial infarction and prediction of infarct size in chest pain patients,” *International Journal of Cardiology*, vol. 114, no. 3, pp. 366–374, 2007.
- [29] J. Ellenius, T. Groth, B. Lindahl, and L. Wallentin, “Early assessment of patients with suspected acute myocardial infarction by biochemical monitoring and neural network analysis,” *Clinical Chemistry*, vol. 43, no. 10, pp. 1919–1925, 1997.
- [30] S. J. Leslie, M. Hartswood, C. Meurig et al., “Clinical decision support software for management of chronic heart failure: development and evaluation,” *Computers in Biology and Medicine*, vol. 36, no. 5, pp. 495–506, 2006.
- [31] B. A. Mobley, E. Schechter, W. E. Moore, P. A. McKee, and J. E. Eichner, “Predictions of coronary artery stenosis by artificial neural network,” *Artificial Intelligence in Medicine*, vol. 18, no. 3, pp. 187–203, 2000.
- [32] S. M. Pedersen, J. S. Jørgensen, and J. B. Pedersen, “Use of neural networks to diagnose acute myocardial infarction. II. A clinical application,” *Clinical Chemistry*, vol. 42, no. 4, pp. 613–617, 1996.
- [33] Z. Shen, M. Clarke, R. Jones, and T. Alberti, “A new neural network structure for detection of coronary heart disease,” *Neural Computing & Applications*, vol. 3, no. 3, pp. 171–177, 1995.
- [34] H. Yan, Y. Jiang, J. Zheng, C. Peng, and Q. Li, “A multilayer perceptron-based medical decision support system for heart disease diagnosis,” *Expert Systems with Applications*, vol. 30, no. 2, pp. 272–281, 2006.
- [35] J. Liu, Z. H. Tang, F. Zeng, Z. Li, and L. Zhou, “Artificial neural network models for prediction of cardiovascular autonomic dysfunction in general Chinese population,” *BMC Medical Informatics and Decision Making*, vol. 13, Article 80, 2013.
- [36] Z. H. Tang, J. Liu, F. Zeng, Z. Li, X. Yu, and L. Zhou, “Comparison of prediction model for cardiovascular autonomic dysfunction using artificial neural network and logistic regression analysis,” *PloS One*, vol. 8, no. 8, Article ID e70571, 2013.
- [37] D. S. Broomhead and D. Lowe, “Multivariable functional interpolation and adaptive networks,” *Complex Systems*, vol. 2, no. 3, pp. 321–355, 1988.
- [38] F. Schwenker, H. A. Kestler, and G. Palm, “Three learning phases for radial-basis-function networks,” *Neural Networks*, vol. 14, no. 4–5, pp. 439–458, 2001.
- [39] P. Dhanalakshmi, S. Palanivel, and V. Ramalingam, “Classification of audio signals using SVM and RBFNN,” *Expert Systems with Applications*, vol. 36, no. 3, pp. 6069–6075, 2009.
- [40] M. Aurélio de Oliveira, O. Possamai, L. V. O. D. Valentina, and C. A. Flesch, “Modeling the leadership-project performance relation: radial basis function, Gaussian and Kriging methods as alternatives to linear regression,” *Expert Systems with Applications*, vol. 40, no. 1, pp. 272–280, 2013.

- [41] S. N. Qasem and S. M. Shamsuddin, "Radial basis function network based on time variant multi-objective particle swarm optimization for medical diseases diagnosis," *Applied Soft Computing Journal*, vol. 11, no. 1, pp. 1427–1438, 2011.
- [42] M. Balasubramanian, S. Palanivel, and V. Ramalingam, "Real time face and mouth recognition using radial basis function neural networks," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6879–6888, 2009.
- [43] S. S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, Englewood Cliffs, NJ, USA, 2007.
- [44] Y. Li, M. J. Pont, and N. Barrie Jones, "Improving the performance of radial basis function classifiers in condition monitoring and fault diagnosis applications where "unknown" faults may occur," *Pattern Recognition Letters*, vol. 23, no. 5, pp. 569–577, 2002.
- [45] Z.-Q. Zhao and D.-S. Huang, "A mended hybrid learning algorithm for radial basis function neural networks to improve generalization capability," *Applied Mathematical Modelling*, vol. 31, no. 7, pp. 1271–1281, 2007.
- [46] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, no. 2, pp. 149–154, 1964.



Hindawi
Submit your manuscripts at
<http://www.hindawi.com>

