33rd Annual International Conference of the IEEE EMBS
Boston, Massachusetts USA, August 30 - September 3, 2011

Use of Radial Basis Functions in Computer-aided Diagnosis of Prostate Cancer

Oscar Marín, Daniel Ruiz, *Member IEEE,* Irene Pérez and Antonio Soriano

***Abstract*— In this paper, we show the results of a study in which we try to test the feasibility of using radial basis functions neural networks (RBFs for short) in clinical decision support systems. We have implemented two instances of RBFs in order to diagnose possible prostate cancer cases from a clinical database. To give an idea about how good the results are, we follow a two-fold approach. On the one hand they are independently evaluated in terms of accuracy, sensitivity and specificity and on the other hand they are compared with the performance over the same database of a classifier widely applied to the medical field problems, as it is multi-layer perceptron (MLP). The experimental results show that RBFs are a useful tool to build up clinical decision support systems.**

I. Introduction

Artificial neural networks (ANNs) are a tool from machine learning theory that has been widely used to solve classification and pattern recognition problems. There are several different implementations of these algorithms, MLPs, RBFs, Learning Vector Quantization networks (LVQs), Self-organizing maps (SOMs), Adaptive Resonance Theory based networks (ARTs), etc [1]. However, as the research on them has advanced, their application has been focused on certain areas depending on their features and the way they perform.

ANNs are able to model complex biological systems by revealing relationships among the input data that cannot always be recognized by conventional analyses [2].

Cancer is a major public health concern in the developed countries. A total of 1,529,560 new cancer cases and 569,490 deaths from cancer were expected to occur in the United States in 2010 [3]. From those, approximately 192,000 men were diagnosed with prostate cancer, and 27,000 of them were expected to die from this disease, what makes prostate cancer the second most common cause of death caused by cancer, among men aged 80 years and older [4].

Manuscript received March 25, 2011. This work has been granted by the Ministerio de Ciencia e Innovación of the Spanish government (Ref. TIN2009-10855) and co-financed by FEDER.

O. Marín is with the Bioinspired Engineering and Health Computing Research Group, University of Alicante, Alicante, P.O. 99 E-03080 Spain (e-mail: omarin@ibisrg.com).

D. Ruiz is with the Computing Technology Department, University of Alicante, Alicante, P.O. 99 E-03080 Spain (e-mail: druiz@dtic.ua.es).

I. Pérez is with the Bioinspired Engineering and Health Computing Research Group, University of Alicante, Alicante, P.O. 99 E-03080 Spain (e-mail: iperez@ibisrg.com).

A. Soriano is with the Computing Technology Department, University of Alicante, Alicante, P.O. 99 E-03080 Spain (e-mail: soriano@dtic.ua.es).

As it is the case of many other kinds of cancer, early detection of prostate cancer symptoms is the best way to treat the disease at its first stages reducing the mortality [5]. The survival rate of prostate cancer soars from 34% when the cancer is detected at the advanced stage to nearly 100% at the early stage [6].

Clinical decision support systems (CDSSs) can be useful to help specialists in the difficult task of diagnosis [7]. A second expert opinion, even if it is from an artificial entity or software acting as a human expert, can support the decision of the doctor. In other cases the clinical decision support system can suggest alternative tests to increase the degree of certainty in a specific diagnosis.

In this project we face the medical problem of prostate cancer diagnosis developing a clinical decision support system that may help the specialist to improve the certainty in the diagnosis avoiding unnecessary biopsies.

After previous researches, and reviewing the literature about the use of different ANNs implementing these systems, we noticed that MLPs have been widely used in researches that imply the use of automated methods on clinical environments, i.e. support the cancer diagnosis [8]–[9]. This is not the case of RBF neural networks that have been barely used for this aim. In fact, most of the researches related to these networks are focused on, developing new training algorithms, testing new radial functions as transference functions, etc. So, there is a noticeable lack of researches on using this type of ANN as a classifier in the clinical field.

Since we think that RBFs features make them suitable for this purpose, we have implemented two different RBF neural networks and used them in a system for supporting doctors diagnosing prostate cancer. Once it is done, we compare the results with previous experiences using MLPs for the same task.

In the next sections we will give in first place an explanation about RBFs: their fundamentals and the motivation behind their use. Next, the experimentation stage is explained, describing the implementation of RBFs and MLP used. Finally, we show the obtained results using RBFs in a prostate cancer diagnosis support decision system.

II. Radial basis functions description

Radial basis function networks were first introduced by Broomhead & Lowe as an alternative to MLPs in 1988, for the aim of making the adjustment of nonlinear functions [9].

RBF's structure is quite similar to MLPs, with an input

and an output layer respectively. However there is a difference in the number of intermediate, or hidden, layers.

RBFs have only one hidden layer while MLPs could have more than one. So, in RBFs we can distinguish 3 different layers (Fig. 1):

- Input layer: where input signals arrive. No processing of them is done at this stage.
- Hidden layer: neurons or nodes within this layer receipt the input signals from the previous layer and perform a local nonlinear transformation over them.
- Output layer: a linear combination of the outputs from the activated nodes in the previous layer is done. This will be the net's output.

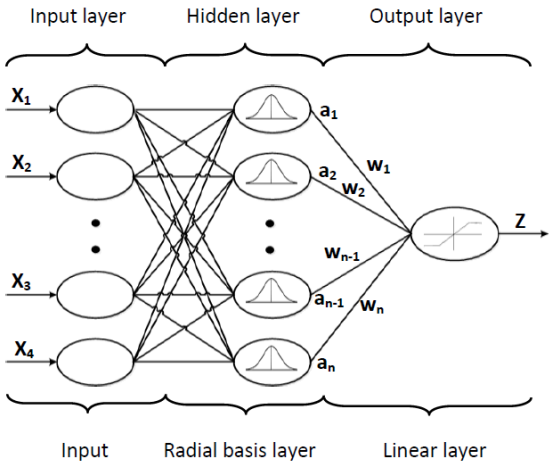

Fig. 1 RBF neural networks architecture.

The nodes within the hidden layer, implement as local transference function a radial basis function that usually is a sort of Gaussian function. By means of these radial functions, each of these nodes contributes to draw a solution region within the global input pattern space. These individual RBF contributions could be depicted as circles (Fig. 2a), and the decision region could be seen as the space result of all circles union, one for each neuron in the hidden layer (Fig. 2b).

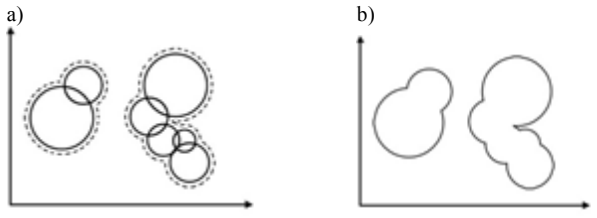

Fig. 2 a) Decision areas of each node. b) Combination of all hidden neuron radial basis function contributions.

The input patterns placed inside these regions, are those that produce a significant net's output. It is easy to see that the use of this kind of functions allows us to build complex solution regions, using only little number of nodes in the hidden layer.

Each node represents a class and each class is represented by two parameters (C_i, s_i). C_i represents the centroid of the class, the middle point in the weights associated to that neuron, where the radial function has its maximum value. On the other hand, s_i is the spread of the radial basis function associated to that neuron, which can be seen as the standard deviation of the radial function.

When the input values arrive to this layer, each node calculates the distance between its centroid and the input patterns values. Let d_i be this value of distance (1).

$$d_i = \sqrt{(x_1 - c_1)^2 + (x_2 - c_2)^2 + \cdots + (x_n - c_n)^2} \qquad (1)$$

The next step is to apply a radial basis function, as a local nonlinear activation function, to the distance values in order to obtain the activation value of each node. Let ϕ be the radial basis function of the node (2), a Gaussian function in our example, and $a_i(n)$ be the the output of the hidden layer (3).

$$\phi(x) = e^{(-x^2/2)} \qquad (2)$$

$$a_{ix} = \phi(d_i/s_i) \quad i = 1,2,\ldots,n \qquad (3)$$

We can notice that, if the input values are similar to the centroid of a hidden neuron, the distance value for this node will be close to 0, and therefore the output of the Gaussian function will be almost 1. The hidden layer and the output layer connections are weighted with a value *w* that will be adjusted by a learning algorithm, usually based on the output's error estimation, like it is the case of nets based only in backpropagation like MLPs. Each processing element in this layer obtains its final value as a linear combination of the previous layer outputs and the weights of the links between the units of both layers, being $Zk(x)$ the output value of each neuron on the output layer, and μ_k a bias value (4).

$$Zk(x) = \sum_{i=1}^{m} w_{ik} a_i(x) + \mu_k, \quad k = 1,2,\ldots,m \qquad (4)$$

In (4) *m* is the number of nodes in the hidden layer and *k* the number of nodes in the output layer.

III. EXPERIMENTS

To assess the performance of RBFs diagnosing prostate cancer, we have implemented two different versions of RBF. Later we tested each one using a diagnostic tests database, evaluating the results independently. Besides this, we compare these independent results with the performance of a previously implemented MLP as a reference.

A. RBFs design

The first version of RBF implemented is a probabilistic neural network (PNN). PNNs are RBFs that behave as the main model previously explained but with a significant difference on the meaning of the output values. The hidden layer computes distances from the input patterns to the

centroid of each neuron. The second layer adds these contributions for each class of inputs to produce, as its net output, a vector of probabilities. Finally, a compete transfer function on the output of the second layer, chooses the maximum of these probabilities, and produces a 1 for that class and a 0 for the other classes. Consequently, the network classifies an input vector into a specific class because that class has the maximum probability of being correct. PNNs have been successfully used in classification problems [10].

The second model of RBF implemented is a general regression neural network (GRNN) [11]. In this case the hidden layer has as many neurons as there are target vectors within the input set. Each neuron's weighted input is the distance between the input vector and its weight vector, and each neuron's net input is the product of its weighted input with its bias. Each neuron's output is its net input passed through a radial basis function. If a neuron's weight vector is equal to the input vector, its weighted input will be 0, its net's input will be 0, and its output will be 1. If the distance between a neuron's weights vector and the input vector equals to the spread value, its weighted input will be the spread, and its output will be 0.5. The second layer also has as many neurons as input/target vectors.

In each case, the network tends to give as an output the target vector that is closer to the input vector in each case.

B. Testing and results analysis

Our clinical database consists of 950 samples from patients who have been tested by expert urologists to check if they suffer from prostate cancer. For 381of these patients, the tests showed that they have prostate cancer. This diagnosis was later confirmed by means of a biopsy study for each patient.

Besides the diagnostic results, the database also contains values for more 14 characteristics for each patient. These characteristics are respectively related with patients personal info (age) and the results for each patient, of tests commonly used by urology experts for prostate cancer diagnosis (Prostate-specific antigen (PSA) in blood level, PSA density, prostate volume, rectal examination results, transitional zone flow, zone transitional volume, intralesional IR, intraprostatic IR, periprostatic IR, state of the prostate capsule, state of the seminal vesicles, quotient, and prostateseminal angle).

Not all the fields are numerical, 5 of them are filled using a subset of medical terms. In order to use these text fields, we have related each term with a number. For example, adenoma, LD nodule, LI nodule and bilateral nodule, which are values of "rectal test results" fields, are translated to 1, 2, 3 and 4 respectively. On the other hand, the diagnosis result has two possible values: "yes" or "no", which we have identified with 1 and 0 respectively and this will be the output of the designed CDSS.

In this work we have divided the input data in two non-overlapping sets. The first one contains the 85% of the input samples, being used to carry out the training and testing processes applying the 10-fold cross validation technique.

The second set consists of the other 15% input samples and it is used to validate the classifier configuration, obtained at the training and test stage.

There are a set of customizable parameters whose best values should be obtained by trial. For this reason, we wrote an executable script to test in a batch way several parameters for PNN and GRNN networks, and compiling metrics after the execution of each one. This process allows us to choose the proper parameters configuration that gives us the best performance rates of each ANN.

Firstly, for the reference MLP, we tried to find the number of hidden layers and the suitable size of each net's layer. We have tested designs that contain 1 and 2 hidden layers with a range from 5 to 20 neurons in each layer. We also looked for the transfer function that will control the input data through the net, testing the tan-sigmoidal, log-sigmoidal, and lineal transfer functions. Other tested parameters were the net's learning rate and the momentum. The configuration that gave the best MLP performance consists of 1 hidden layer of 13 neurons and 1 on the output layer, using the tan-sigmoidal as a transfer function with a learning rate value of 0.2 and a momentum of 0.3, and performing a training process of 10000 epochs, Table I.

TABLE I
Reference MLP configuration parameters

Layers	Neurons	Transfer Function	Training Algorithm	Epochs
2	[13 1]	Tangent-sigmoid	Levenberg-Mardquardt	10000

For the RBFs we tested different parameters combinations that include the spread and ridge values of the radial functions, the minimum standard deviation, and the number of neurons inside the hidden layer, Table II.

TABLE II
PNN and GRNN configuration parameters

	Layers	Neurons	Radial Fnct.	Ridge	Spread	Stdrd. Dev.	Epochs
GRNN	2	[10 1]	$e^{(-x^2/2)}$	1.0E-8	0.1	0.1	5000
PNN	2	[10 1]	$e^{(-x^2/2)}$	1.0E-8	0.1	0.1	3000

Finally we use the testing set to evaluate each classifier separately. We are interested not only in the accuracy rate of each classifier, but also in sensitivity and specificity rates, since we are performing a diagnostic task within a CDSS. The results from the execution of each classifier can be seen in Table III.

TABLE III
Results (in percentage) of the testing executions

	MLP	GRNN	PNN
Accuracy	71.05	81.05	85.26
Sensitivity	46.96	57.57	74.24
Specificity	83.87	93.54	91.12

On Table III we can see that both RBFs obtain better rates of accuracy than MLP, but the main improvement achieved using these nets is the growth in the rates of specificity and sensitivity. These are remarkable achievements because they have a two-fold direct consequence: an improvement of patients' quality of life and a reduction of the costs that are associated with diagnostic and treatment tasks. Since we are dealing with prostate cancer diagnostic data, the improved specificity and sensitivity rates give an idea about the reduction of the needed number of biopsies to be performed due to misclassification of healthy patients, and the reduction of people who have prostate cancer and are wrongly diagnosed, respectively.

Analyzing all the measures together, we can see that PNN is the most suitable classifier, among the tested ones, to implement our prostate cancer decision support system, despite of not obtaining the best values for all the measured rates.

IV. Conclusion

The goal of this paper is to evaluate the utility and the performance of RBF neural networks, within a clinical environment, as a basis of a CDSS for prostate cancer diagnosis. We have chosen two different RBFs algorithms, a probabilistic neural network and a generalized regression neural network, to carry out the experimentation stage in order to test as much features of these algorithms as possible. These ANNs have been trained and tested using a medical database with information from clinical tests applied to patients with suspicions of suffering from prostate cancer.

We collected statistical measures of the performance of the implemented RBFs to evaluate them, and from a MLP as a reference of a well-known and widely applied to diagnostic tasks classifier. Not only accuracy, but also sensitivity and specificity rates (both are better in RBF neural networks) prove that RBF neural networks, although their design can be more complex, are an interesting alternative to have in mind in the design of CDSSs.

As future work, we will continue studying about RBF neural networks and their application as classifiers in CDSSs and, in particular, in prostate cancer diagnosis. We would also like to compare RBF neural networks performance to other artificial intelligence techniques for classifying like SVMs, or classifier assembling methods as boosting. Our final objective is to develop a CDSS for aiding in the diagnosis of prostate cancer with high rates of accuracy and reliability.

References

[1] S. Haykin, *Neural Networks and Learning Machines,* 3rd. ed. Upper Saddle River, New Jersey: Prentice Hall, 2008.

[2] H. Shin, M. K. Markey, "A machine learning perspective on the development of clinical decision support systems utilizing mass spectra of blood samples". *Journal of Biomedical Informatics*, 2006.

[3] A. Jemal, R. Siegel, J. Xu, E. Ward, "USA Cancer Statistics, 2010". *CA Cancer Journal for Clinicians*, 2010.

[4] M. D. Wolf, R. C. Wender, R. B. Etzioni, I. M. Thompson, A. D. D'Amico, "American Cancer Society Guideline for Early Detection of Prostate Cancer: Update 2010". *CA Cancer Journal for Clinicians*, 2010.

[5] O. W. Brawley, D. P. Ankerst, I. M. Thompson, "Screening for Prostate Cancer". *CA Cancer Journal for Clinicians*, 2009.

[6] M. K. Brawer, "Prostate-specific antigen: Current status", *CA Cancer Journal for Clinicians*, 1999.

[7] E. Coiera, *Guide to Health Informatics*. London, UK: Hodder Arnold, 2003.

[8] P. J. Lisboa, A. F. G. Taktak, "The use or artificial neural networks in decision support in cancer: A systematic review" *Neural Networks*, vol. 19, pp. 408-415, 2006.

[9] M. Çinar, E. Z. Engin, Y. Z. Egin, "Early prostate cancer diagnosis by using artificial neural networks and support vector machines" *Expert Systems with Applications*, vol. 36, sup. 3, pp. 6357-6361, 2009.

[10] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, "Training neural network classifiers for medical decision making: The effects of imbalanced data sets on classification performance" *Neural Networks*, vol. 21, sup. 2-3, pp. 427-436, 2008.

[11] H. Wang, H. Wong, H. Zhu, T. T. C. Yip, "A neural network-based biomarker association information extraction approach for cancer classification" *Journal of Biomedical Informatics*, vol. 42, sup. 4, pp. 654-666, 2009.

[12] A. Castaño, C. Hervás-Martínez, P. A. Gutierrez, F. Fernández-Navarro, M. M. García, "Classification by Evolutionary Generalized Radial Basis Functions" in: *9th IEEE International Conference on Intelligent Systems Design and Applications*, 2009, pp. 203–208.

[13] D. S. Broomhead, D. Lowe, "Radial Basis Functions, Multi-Variable Functional Interpolation and Adaptive Networks" *Royal Signals and Radar Establishment,* 1988.

[14] Ö. Polat, T. Yildirim, "Patterns Recognition Without Feature Extraction Using Probabilistic Neural Networks" in *International Conference on Intelligent Computing in Signal Processing and Pattern Recognition*. Intelligent Computing, 2006, pp. 402-408.

[15] Ö. Polat, T. Yildirim, "Recognition of Patterns Without Feature Extraction by GRNN" in *8th International Conference on Adaptive and Natural Computing Algorithms*, part II, 2007, pp. 161-168.