The Separation of Radar Clutters using Multi-Layer Perceptron

Mohammad Akhondi Darzikolaei*
Faculty of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran
m.akhondi@stu.nit.ac.ir
Ata Ebrahimzade
Faculty of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran
E_zade@nit.ac.ir
Elahe Gholami
Faculty of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran
e.gholami8869@yahoo.com

Received: 15/May/2016 Revised: 31/Dec/2016 Accepted: 31/Jan/2017

Abstract

Clutter usually has negative influence on the detection performance of radars. So, the recognition of clutters is crucial to detect targets and the role of clutters in detection cannot be ignored. The design of radar detectors and clutter classifiers are really complicated issues. Therefore, in this paper aims to classify radar clutters. The novel proposed MLP-based classifier for separating radar clutters is introduced. This classifier is designed with different hidden layers and five training algorithms. These training algorithms consist of Levenberg-Marquardt, conjugate gradient, resilient back-propagation, BFGS and one step secant algorithms. Statistical distributions are established models which widely used in the performance calculations of radar clutters. Hence In this research, Rayleigh, Log normal, Weibull and K-distribution clutters are utilized as input data. Then Burg's reflection coefficients, skewness and kurtosis are three features which applied to extract the best characteristics of input data. In the next step, the proposed classifier is tested in different conditions and the results represent that the proposed MLP-based classifier is very successful and can distinguish clutters with high accuracy. Comparing the results of proposed technique and RBF-based classifier show that proposed method is more efficient. The results of simulations prove that the validity of MLP-based method.

Keywords: Clutter; Classifier; Feature; Neural Network; Radar.

1. Introduction

The term radar is an abbreviation for radio detection and ranging. The rudimentary concept of radar system is inspired by echolocation animals such as bats and dolphins. Radar is a system which detects, locates and measures the speed of objects using echo electromagnetic waves. It transmits electromagnetic waves into environments and receives the echoes from objects. It is apparent that radar system is effected by progression of modern technology. This improvement makes the analysis of radar performance more complicated. Many negative factors can have destructive influences on radar performances. Clutter has really the most negative role on radar echo signals. Clutter is any unwanted signal which can disorder echoes from radar. Clutter can be reflected from any things such as lands, sea, forests, mountains and weather conditions. Because of stochastic and variable nature of clutters, radar specialists usually apply probability density functions for describing the traits of clutters. Gaussian, Weibull, Rayleigh, K-distribution and log-normal are most popular and widely used models. Adaptive techniques for detection, tracking and classification of clutters are very crucial. Artificial Neural Networks and Heuristic Algorithms have been used for radar signal processing which requires high capacity to match with different conditions. [1], [2], [3]

In the field of soft computing, Artificial Neural Networks [1] is one of the most important methods. Its invention is inspired by the neurons of human brain. Mc culloch and Pitts [4] were the first ones modeled mathematically the neural networks. The simplicity, low computational cost and high performance are some significant characteristics of this computational approach. Feed forward Neural Networks [5] are very popular tools among other kinds of Neural Networks. They receive data as inputs on one side and prepare outputs from other side by connections between neurons in various layers. Multi-layer perceptron [2] [6,7] is the one type of feed forward neural networks. MLP has more than one perceptron in different layers. This helps it to solve nonlinear problems. Pattern classification [8], data prediction [9], pattern recognition, remote sensing and optimization are few applications of MLPs. The amazing trait of MLPs is learning [10]. Similar to human brain, MLPs have aptitude to learn from experiences. This part is common in all neural networks. Back Propagation algorithms are the instances of learning algorithms which are widely used with MLPs.

As mentioned, Neural Networks and Soft computing algorithms have been used successfully for radar signal

[1] ANN
[2] MLP

* Corresponding Author

processing. Authors in [11] classified various kinds of clutters. They tried to categorize birds, weather and ground clutters. Their data was obtained from Air Traffic Control. In fact, data were experimental were included the amplitude and phase of echo signals. Haykin et al in this reference extracted a set of best features which can differentiate among different clutter models. In reference [12] the radar target detection was done with Artificial Neural Network. Authors used Prony's algorithm to extract time-domain target features. Multi-Layer Perceptron and the Self Organizing Maps were utilized. These networks had been tested and each network had been trained on a wide range of SNR and with various data to appraise the training invariant traits of each network. Authors in [13] considered radar signal detection using Artificial Neural Networks in just K-distributed environment. They tested two training algorithms. Back propagation and Genetic algorithms for an MLP structure were used. In [14] authors present the problem of detecting targets in simulated land clutter. The clutter was modeled by Weibull distribution. Authors in this reference were tried to find a detection scheme to determine the target position easily. Because high-level clutter echoes were received, they proposed detection scheme based on Neural Network, where feedforward multilayer perceptron was used. Then, they compared their proposed scheme with a coherent detector commonly used for Weibull-distributed clutter and concluded the performance improvement achieved by using their proposed method. Reference [15] described the classification of radar returns Sea and ground clutters. Authors first explained an analysis of radar clutter data to validate the K amplitude distribution and the autoregressive modelling of the spectrum. Then, they briefly introduced a multi-layer neural network classifier. The Neural Network inputs were included the shape parameter of the K-distribution, the magnitude and the phase of the poles and the reflection coefficients which were calculated by using the Burg's algorithm. In [16] authors aimed to improve the detection radar performance in presence of snow clutter. They extracted suitable features which were occupied to separate targets and snow clutter. A Bayes classifier, a multilayer perceptron and a radial basis function network were tested and compared. In this paper, we classify the different types of radar clutters. The general procedure of this classification is shown in Fig.1.

Fig. 1. the general procedure of classification of radar clutters

In This paper, we use MLP method to classify four different kinds of clutters. We use five different training algorithms to form our neural networks. Using four clutters and different training algorithms were novel concepts and it was not done in any papers. We compared MLP and RBF to show that MLP is more suitable for clutter data. Fortunately the results prove this issue. These Three characteristics are the preferences of our paper.

This paper has following procedure: first radar clutter models which they are used as input data are introduced. Rayleigh, Log normal, Weibull and K-distribution clutters are modeled. In section 3, three suitable features for clutter data are described. Burg's reflection coefficients, Skewness and kurtosis are these three features. In next part, MLP as classifier is explained and 5 training algorithms for MLP-based classifier is described. Section 5 represents some results of simulations. These results show the validity of proposed MLP-based classifier. In this part also the results of simulations are compared with results of RBF-based classifier. In the last part, the conclusions of simulations and this research are mentioned.

2. Radar Clutter Models

This section introduces radar clutter models. These models can be used for sea and land and weather. Because clutters are variable and random echoes, statistical distributions are used to describe the characteristics of clutters.

2.1 Rayleigh Distribution

The probability density distribution function of Rayleigh distribution [17] is

$$f(x) = \frac{x}{\sigma_v^2} exp\left(-\frac{x^2}{2\sigma_v^2}\right) \quad x \geq 0 \qquad (1)$$

Where x is clutter amplitude, σ_v is standard deviation of clutter. The distribution function is

$$F(x) = 1 - exp(-\left(\frac{x}{\sigma_v}\right)^2) \qquad (2)$$

In order to well describe the relationship of parameter σ_v and environment, let $\sigma_v = \frac{\sigma}{\lambda_0}$ into (1), we can get:

$$f(x) = \frac{x\lambda_0}{\sigma^2} exp\left(-\frac{x^2\lambda_0^2}{2\sigma^2}\right), \quad x > 0 \qquad (3)$$

Where λ_0 is radar wavelength. Fig. 2 shows Rayleigh distribution with different parameters.

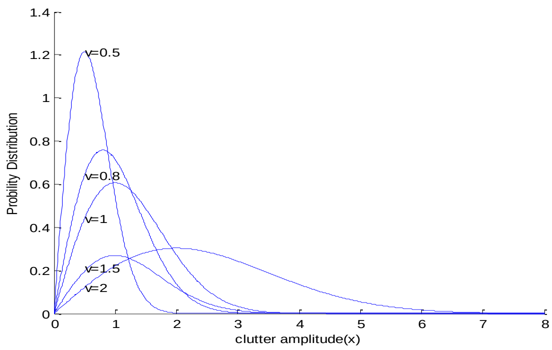

Fig. 2. Rayleigh PDF with different parameters

2.2 Weibull Distribution

Weibull distribution is used usually for modeling of clutter in low grazing angle.it can be used as weather, sea and land clutter. Weibull density function is:

$$f(x) = \frac{\beta x^{\beta-1}}{c^{\beta}} exp\left(\left(\frac{-x}{c}\right)^{\beta}\right) \quad ; \quad x \geq 0 \qquad (4)$$

Where β is shape parameter and c is scale parameter. Weibull distribution with different parameters is shown in Fig.3.

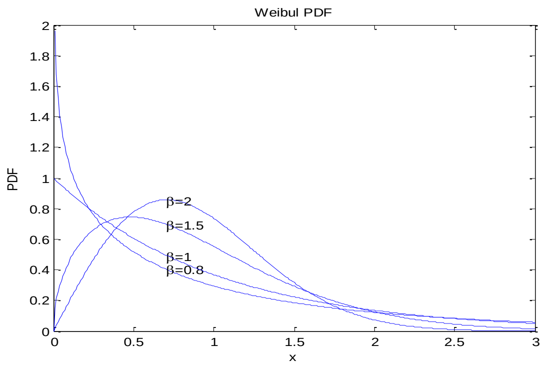

Fig. 3. Weibull PDF with different parameters

2.3 K-distribution

This probability distribution for modeling the statistics of clutter is described as a compound distribution that consists of two components the local power and the speckle component. The K-distribution [18] probability density function describing amplitude x is

$$f(x) = \frac{2b^{\frac{v+1}{2}} x^{\frac{v-1}{2}}}{\Gamma(v)} K_{v-1}(2\sqrt{bx}) \qquad (5)$$

This is characterized by a scale parameter, b, and a shape parameter, v. In Fig. 4 K-distribution with different shape parameters is shown.

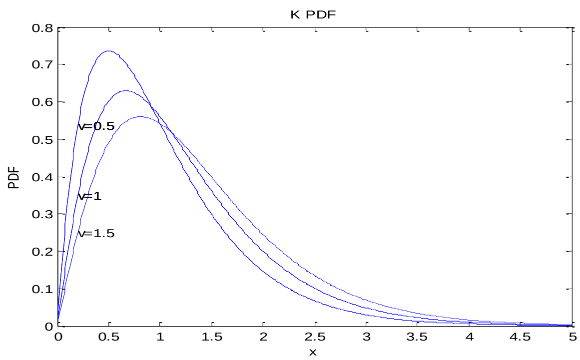

Fig. 4. K-distribution PDF with different shape parameters

2.4 Log Normal Distribution

One of the first models used to describe non Rayleigh clutters was Log normal [19] because it has longer tail than Rayleigh. In the Log normal probability density function the clutter echo power which is expressed in decibel (dB) is Gaussian. The log normal probability density function is:

$$f(x) = \frac{1}{\sqrt{2\pi}sx} exp[\frac{1}{2s^2}(ln\, x/x_m)^2] \; ; \; x \geq 0 \qquad (6)$$

Where s is standard deviation and x_m is average of x.

Fig.5 shows the variations of log normal pdf when ‘s’ changes and $x_m = 1$.

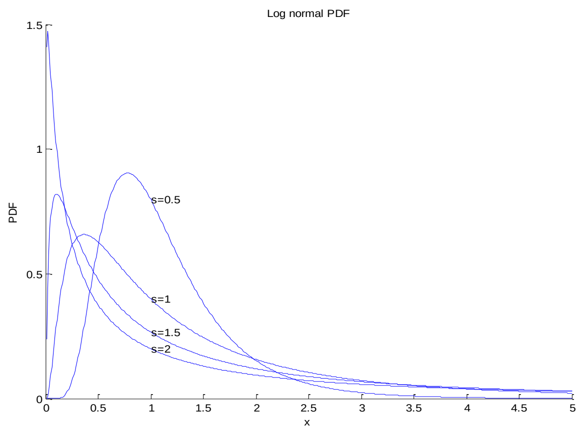

Fig. 5. Log normal PDF with’ x_m=1’and different values of ‘s’

3. Feature Extraction

Feature extraction plays an essential role in each classification problems. It is necessary to extract the set of features which can be applied to distinguish the members of input types. The study of clutter statistical features has been performed to recognize the most suitable set to be used as classifier inputs. This is suitable way for controlling the computational cost and improving the capabilities of classifiers. In this paper we have considered four statistical distributions such as Log normal, Rayleigh, Weibull and K-distribution for classification. We use three features for radar clutters. Skewness and kurtosis as high order moments and Burg reflection coefficients which are described below.

3.1 Burg Reflection Coefficient

Burg’s reflection coefficients [20] are utilized as spectral features for clutters. These coefficients are obtained from maximum entropy method (MEM) of spectral analysis [21].

The coefficients arise out of the lattice implementation of the prediction error filter (PEF) which attempts to minimize the prediction error power at each stage. This minimization results in a whitening filter and as such the reflection coefficients represent incremental predictable information extracted from the time series at each stage. Therefore we use burg’s reflection coefficients to extract the best features. These coefficients are defined as [22]:

$$\rho_m = \frac{-2\sum_{i=1}^{L}\sum_{k=m+1}^{K} f_{m-1,i}(k)\, b^{*}_{m-1}(k-1)}{\sum_{i=1}^{L}\sum_{k=m+1}^{K}\left(|f_{m-1,i}(k)|^2+|b_{m-1,i}(k-1)|^2\right)} \qquad (7)$$

Where $f_{m,i}(k)$ and $b_{m,i}(k)$ are the forward and backward prediction errors. They are obtained as:

$$f_{m,i}(k) = f_{m-1,i}(k) + \rho_m b_{m-1,i}(k-1) \qquad (8)$$

$$b_{m,i}(k) = b_{m-1,i}(k-1) + \rho_m^* f_{m-1,i}(k) \quad (9)$$

The asterik in equations (7) and (9) represent complex conjugation. For a specified index i, the prediction errors $f_{0,i}(k)$ and $b_{0,i}(k)$ are intialized with the input data as follows:

$$f_{0,i}(k) = b_{0,i}(k) = x_i(k) \quad (10)$$

For each of the L lattice filters, the index i implies to i^{th} time series part of length K. m^{th} part of the prediction-error filter is also determined with m and the k^{th} sample in a time series is shown with k.

Although Burg's reflection coefficients are features which are used for phase of clutter data but it is common to use magnitude of $|\rho_1|$ directly as the feature of amplitude of clutter data. So in this paper, we use $|\rho_1|$ as feature for our clutter data, because our input data are amplitude of log normal, Rayleigh, weibull and K-distribution.

3.2 Skewness and Kurtosis

In addition to mathematical description, Skewness and kurtosis have physical meanings [23]. The skewness represents the asymmetry of the distribution from its mean and kurtosis measures how peaky or flat with respect to Gaussian distribution.

In this paper we apply these two high order moment as the feature of amplitude of radar clutters. They are defined as:

$$Skewness(x) = \sum_M \frac{(x-\mu(x))^3}{\sigma^3} \quad (11)$$

$$Kurtosis(x) = \sum_M \frac{(x-\mu(x))^4}{\sigma^4} \quad (12)$$

Where μ refers to average and σ shows the standard deviation of x.

4. Classifiers

This section explains the classifiers which are used in this paper.

4.1 Multi layer Perceptron Network (MLP)

In this paper, we have utilized MLP Neural Networks as classifiers. An MLP neural network includes various layers. An input layer of source nodes, one or more hidden layers of computation nodes (neurons) and output layers. It should be mentioned that each layer is fully connected to next one. Inputs are spread through the network layer by layer and MLP gives a non-linear mapping of the inputs at output layers.Fig.6 shows MLP topology [24]. The recognition basically consists of two training and testing phases. In training stage, weights are calculated according to the chosen learning algorithm. Learning algorithms and their speeds are very essential problems for MLP. The aim of training is to minimize the global error (E) which is defined as:

$$E = \frac{1}{P}\sum_{p=1}^{P} E_p \quad (13)$$

Where P is the total number of training patterns and E_p is the error for training pattern *(p)*. E_p is obtained by below formula:

$$E_p = \frac{1}{2}\sum_{i=1}^{N}(o_i - t_i)^2 \quad (14)$$

Where N is the total number of output neurons, o_i is the network output at the i^{th} output neuron and t_i is the target output at the i^{th} output neuron. In every training algorithm, the objective is to decrease this global error by adjusting the weights and biases.

One of the popular learning algorithm is Back Propagation (BP) [25]. In BP, a simple gradient descent algorithm updates weight values by using following formula[26]:

$$w_{k+1} = w_k - a_k\, g_k \quad (15)$$

Where, g_k is the current gradient, a_k is learning rate and w_k is the vector of current weights.

Although BP is still popular, in some conditions, BP network classifiers generate non-robust responses and converge to local mininmum. New algorithms have been introduced for training stage. However, some algorithms needs much computing power to get good training especially when dealing with a large training set.

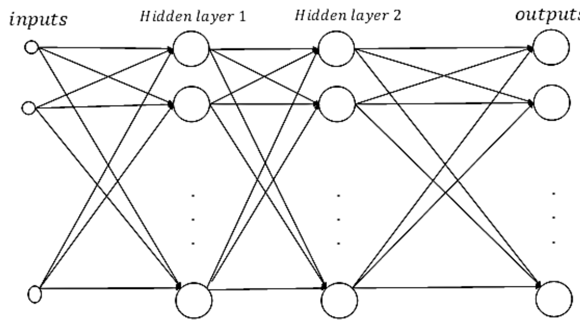

Fig. 6. MLP topology [24]

In this paper following learning algorithms are considered.

4.1.1 Levenberg-Marquardt Algorithm

The objective of designing Levenberg-Marquardt (LM) algorithm was to achieve second order training speed without having to calculate the Hessian matrix [26]. When the performance function has the form of a sum of squares, then then Hessian matrix can be approximated from following formula:

$$H = J^T J \quad (16)$$

And also gradient is calculated as:

$$g = J^T e \quad (17)$$

Where J is the Jacobian matrix, which cosists first derivatives of the network errors with respect to the weights and biases, and e is a vector of network errors.

The Levenberg-Marquardt algorithm [27] utilizes this approximation to the Hessian matrix in the following Newton-like update:

$$w_{k+1} = w_k - [J^T J + \mu I]^{-1} J^T e \tag{18}$$

Where J is jacobian matrix, e is a vector of network errors and μ is a constant.

4.1.2 Conjugate Garadient Algorithm

The basic back propagation algorithm controls the weights in the steepest descent direction (the most negative of the gradients). This is the direction in which the performance function is declining very quickly. Though the function declines most swiftly along the negative gradient, this does not necessarily generate the fastest convergence. In the conjugate gradient algorithms, searching is done along conjugate directions, which converge faster than steepest descent directions.

Conjugate gradient algorithms commence by searching in the steepest descent direction on the first iteration.

$$p_0 = -g_0 \tag{19}$$

A line search is then done to choose optimal distance to move along the current search direction:

$$w_{k+1} = w_k + a_k p_k \tag{20}$$

Then the next search direction is chosen so that it is conjugate to previous search directions. Combining the new steepest descent direction with the previous search direction is popular method for determining the new search direction:

$$p_k = -g_k + \beta_k \, p_{k-1} \tag{21}$$

The way in which β_k is computed with Fletcher-Reeves update as:

$$\beta_k = \frac{g_k^T \, g_k}{g_{k-1}^T \, g_{k-1}} \tag{22}$$

Above formula is the ratio of the norm squared of the current gradient to norm squard of the previous gradient [28].

4.1.3 Resilient Back-Propagation (RPROP) Algorithm

RPROP considers the sign of derivatives as the indication for the direction of the weight update [29]. In doing so, the size of the partial derivative does not influence the weight step. The following equation shows the adaptation of the update values of Δ_{ij} (weight changes) for the RPROP algorithm. For initialization, all Δ_{ij} are set to small positive values:

$$\Delta_{ij}(t) = \begin{cases} \eta^+ * \Delta_{ij}(t-1)\,; & if \frac{\delta E}{\delta w_{ij}}(t-1)\,\frac{\delta E}{\delta w_{ij}}(t) > 0 \\ \eta^- * \Delta_{ij}(t-1)\,; & if \frac{\delta E}{\delta w_{ij}}(t-1)\,\frac{\delta E}{\delta w_{ij}}(t) < 0 \\ \eta^0 * \Delta_{ij}(t-1)\,; & otherwise \end{cases} \tag{23}$$

Where $\eta^0 = 1$, $0 < \eta^- < 1 < \eta^+$ and $\eta^{-,0,+}$ are known as the update factors. Whenever the derivative of the corresponding weight changes its sign, it implies that the previous update value is too large and it has skipped a minimum. Therefore, the update value is then reduced η^- as shown above. However, if the derivative retains its sign, the update value is η^+ increased. This will help to accelerate convergence in shallow areas. To avoid over- acceleration, in the epoch following the application of η^+, the new update value is neither increased nor decreased (η^0) from the previous one. Note that the values of Δ_{ij} remain non-negative in every epoch. This update value adaptation process is then followed by the actual weight update process, which is governed by the following equations:

$$\Delta_{ij}(t) = \begin{cases} -\Delta_{ij}(t)\,; & if \frac{\delta E}{\delta w_{ij}}(t) > 0 \\ +\Delta_{ij}(t)\,; & if \frac{\delta E}{\delta w_{ij}}(t) < 0 \\ 0\,; & otherwise \end{cases} \tag{24}$$

Weight values are updated with below formula:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta_{ij}(t) \tag{25}$$

The update values and weights are changed every time the whole pattern set has been presented once to the network (learning by epoch).

4.1.4 BFGS Algorithm

Newton's method is an alternative to the conjugated gradient methods for fast optimization. The basic step of Newton's method is:

$$w_{k+1} = w_k - A^{-1} g_k \tag{26}$$

Where A_k is the Hessian matrix of performance index at the current values of the weights. Because of high computational cost of the Hessian matrix, usually some of algorithms which don not need to the computation of second derivatives are introduced. These are called Quasi-Newton (or secant) method. The quasi-Newton method, which has been most successful in published studies, is the Broyden, Fletcher, Goldfarb and shanno (BFGS) update [30].

4.1.5 One Step Secant (OSS) Algorithm

The one step secant (OSS) method is an attempt to bridge the gap between the conjugate gradient algorithms and the quasi-Newton algorithms. OSS algorithm does not save the complete Hessian matrix, it assumes that at each iteration, the previous Hessian was the identity matrix.

4.2 Radial Basis Function (RBF) Network

Radial basis function neural networks (RBFN) are popular tools for multivariate approximation, time series forecasting, image processing, speech recognition, classification and etc., since their properties of localization, robustness and stability [31]. The basic structure of a RBFN is a two layer, feed-forward network in which the activation functions of the neurons of the hidden layer are radial basis functions (RBF). Each hidden neuron calculates the distance from its input to the neuron's central point, *c*, and applies the RBF to that distance. The neurons of the output layer perform a weighted sum between the outputs of the hidden layer and weights of the links that connect both output and hidden layer, in other words linear function is existed between the hidden layer and the output layer:

$$h_i(x) = \phi\left(\frac{\left||x-c_i|\right|^2}{r_i^2}\right) \tag{27}$$

$$f_j(x) = \sum w_{ij} h_i(x) + w_0 \quad (28)$$

where x is the input, ϕ is the RBF, c_i is the center of the i^{th} hidden neuron, r_i is its radius, w_{ij} is the weight links that connect hidden neuron number i and output neuron number j, and w_0 is a bias for the output neuron.

The problem of automatic RBFN design is an important subject. The original regularization RBF theory, proposed that the number of basis functions should be equal to the number of training samples. The basis functions are centered on the training samples and the only unknown parameters are the linear weights, which can be determined efficiently by solving the system of linear equations. However, the resulting networks are complex and often ill-conditioned. Generalized RBFNs are designed with fewer nodes than there are samples in the training set, which results in less complex networks. However, the number of basis functions, their centers and widths, have to be determined. In this paper we have proposed an efficient method based on evolutionary RBF neural networks by implementing improved bees algorithm. The aim of this model is to fit a given data set with sufficient accuracy, and more importantly, generalizes well to unseen data while the neural network is maintaining a reasonable size.

5. Simulation Results

This section represents some of the simulation results of the proposed method for classification of radar clutters. We have used MATLAB as simulator.

In this paper, we design a classifier by using artificial neural networks. The first step in this classifier is producing suitable data set. As mentioned, our input data includes radar clutters. Because our works were done in university and educational environment, we do not access to real and experimental clutter data. So, like other papers, we generate clutters with SIRP and ZMNL methods. These two methods are very common for generate clutter data. We produce 12000 clutter patterns. Log normal, Rayleigh, Weibull and K-distribution were four clutters which are simulated. Our input data were included:

a) Log normal: 3000 patterns with unity mean and standard deviation equal to 0.9.

b) Rayleigh: 3000 patterns with unity standard deviation

c) Weibull: 3000 patterns which its shape parameter is 1.8 and scale parameter is 1.2.

d) K-distribution: 3000 patterns with shape parameter equal to 2 and unity scale parameter.

After generation of input data, we extract their suitable features. In this simulation, we use skewness and kurtosis and $|\rho_1|$ as feature for clutter data.

Then we give these suitable input data to classifier based on MLP neural networks. We design various MLP classifier with different number of hidden layers and neurons. Another characteristic of our work is that we train our data with different learning algorithms. Table 1 shows the list of algorithms which we use for training. In all conditions, we choose 8000 patterns of input data as training data and others 4000 patterns as test data. Since we prove that our proposed method is valid and has high accuracy, we compare it with RBF neural network. All results show that the proposed method has high accuracy.

Table 1: Five different training algorithms used for training of MLP neural networks

Algorithm	Acronym
Resilient	RP
Scaled Conjugate Gradient	SCG
Broyden, Fletcher, Goldfarb and Shanno (BFGS) Quasi-Newton	BFGS
One Step Secant Quasi-Newton	OSS
Levenberg-Marquardt	LM

Because the results of artificial neural network are random, we repeat simulations for 10 iterations and put the average of values in following tables. Note that all of values in tables are respect to percentage. In these experiments, we test our proposed MLP-based classifier in various conditions. First, for different hidden layers which have various neurons we have examined our classifier. Theses simulation were done in four cases. In first case, classifier has two hidden layers which one of them has 20 neurons and another has 15 neurons. In second condition, MLP-based classifier also has two hidden layers but first layer has 30 and second one has 15 neurons. In two other cases, we design one hidden layer for classifier and in each layer are 25 or 35 neurons. All of these cases were repeated for 5 different training algorithms and MLP was trained by LM, RP, OSS, SCG and BFGS training algorithms. The results of all these four cases are shown in tables 2 to 5.

Table 2. confusion matrix of proposed MLP-based classifier with 2 hidden layers and (20*15) neurons

Training algorithm		Rayleigh	Log normal	weibull	K dist.
LM	Rayleigh	**98.6%**	0	0.4%	1%
	Log normal	0	**93.2%**	6.8%	0
	weibull	0.6%	6.4%	**93%**	0
	K dist.	0.2%	0	0	**99.8%**
RP	Rayleigh	**91.8%**	2.3%	0	5.9%
	Log normal	0	**93.5%**	6.5%	0
	weibull	0	9%	**89.7%**	1.3%
	K dist.	0	0.1%	1.6%	**98.3%**
BFGS	Rayleigh	**94%**	0	0	6%
	Log normal	0	**90%**	10%	0
	weibull	0.7%	10.1%	**89.2%**	0
	K dist.	1.8%	0.5%	0	**97.7%**
OSS	Rayleigh	**99%**	0	0	1%
	Log normal	0	**96.3%**	3.7%	0
	weibull	0	8%	**90.7%**	1.3%
	K dist.	0.1%	0	2.3%	**97.6%**
SCG	Rayleigh	**99.1%**	0	0	0.9%
	Log normal	0	**93.7%**	6.3%	0
	weibull	0	9.3%	**90.4%**	0.3%
	K dist.	0.4	0	0.7	**98.9%**

Table 3. confusion matrix of proposed MLP-based classifier with 2 hidden layers and (30*15) neurons

Training algorithm		Rayleigh	Log normal	weibull	K dist.
LM	**Rayleigh**	**95.4%**	0	0	4.6%
	Log normal	0	**94.5%**	5.5%	0
	weibull	0.1%	6.4%	**93.5%**	0
	K dist.	0	0.3%	0.5%	**99.2%**
RP	**Rayleigh**	**95.2%**	0	1.2%	3.6%
	Log normal	0	**94.1%**	5.9%	0
	weibull	0	9.2%	**90.8%**	0
	K dist.	1.2%	0	0.7%	**98.1%**
BFGS	**Rayleigh**	**99.3%**	0	0	0.7%
	Log normal	0	**96.3%**	3.7%	0
	weibull	0	7.9%	**92.1%**	0
	K dist.	0	1.4%	0	**98.6%**
OSS	**Rayleigh**	**99.8%**	0	0.2%	0
	Log normal	0	**96.4%**	3.6%	0
	weibull	0.1%	6.4%	**93.5%**	0
	K dist.	0.2%	0.4%	0	**99.4%**
SCG	**Rayleigh**	**99.1%**	0.9%	0	0
	Log normal	0	**92.5%**	7.5%	0
	weibull	0.7%	8.2%	**91.1%**	0
	K dist.	0	0.1	0.2	**99.7%**

Table 4. confusion matrix of MLP classifier with 1 layer and 25 neurons

Training algorithm		Rayleigh	Log normal	weibull	K dist.
LM	**Rayleigh**	**91.9%**	0	6.2%	1.9%
	Log normal	0	**92.4%**	7.6%	0
	weibull	0.1%	7.3%	**89.2%**	3.4%
	K dist.	0	0.6%	0.7%	**98.7%**
RP	**Rayleigh**	**98.1%**	0	0	1.9%
	Log normal	0	**97.8%**	2.2%	0
	weibull	1%	8.4%	**90.6%**	0
	K dist.	6.8	0	0	**93.2%**
BFGS	**Rayleigh**	**99.2%**	0	0	0.8%
	Log normal	0	**98.8%**	1.2%	0
	weibull	1.6%	0	**90.2%**	8.2%
	K dist.	0	2.3%	0.4%	**97.3%**
OSS	**Rayleigh**	**99.4%**	0	0.1	0.5%
	Log normal	0	**96.1%**	3.9%	0
	weibull	1.2%	5.3%	**93.5%**	0
	K dist.	0	0.4%	1.6%	**98%**
SCG	**Rayleigh**	**99.8%**	0.2%	0	0
	Log normal	0	**95%**	5%	0
	weibull	0	6.6%	**92.6%**	0.8
	K dist.	4.3%	0.3%	0	**95.4%**

Table 5. confusion matrix of proposed MLP-based classifier with 1 hidden layer and (35) neurons

Training algorithm		Rayleigh	Log normal	weibull	K dist.
LM	**Rayleigh**	**98.4%**	0	0	1.6%
	Log normal	0	**93.6%**	6.4%	0
	weibull	1.1%	5.6%	**93.3%**	0
	K dist.	0	2.3%	0	**97.7%**
RP	**Rayleigh**	**96.3%**	0	0	3.7%
	Log normal	0	**95.4%**	4.6%	0
	weibull	0.5%	7.4%	**92.1%**	0
	K dist.	0	0	1.2	**98.8%**
BFGS	**Rayleigh**	**99.8%**	0	0	0.2%
	Log normal	0	**94.7%**	5.3%	0
	weibull	0	8.3%	**91.1%**	0.6%
	K dist.	3.6%	0.6%	0	**95.8%**
OSS	**Rayleigh**	**98.8%**	0.8%	0.4%	0
	Log normal	0	**96.6%**	3.2%	0.2%
	weibull	8.4%	0	**89.3%**	2.3%
	K dist.	0.6%	0.8%	0	**98.6%**
SCG	**Rayleigh**	**99.2%**	0	0	0.8%
	Log normal	0	**96%**	4%	0
	weibull	4.8%	3.4%	**91.8%**	0
	K dist.	0	1.6%	5%	**97.9%**

It is realizable from tables 2 to 5 that the proposed MLP-based classifier can separate radar clutters very successfully. Because in each case, the percentage of recognition of output is very high and all of them are almost more than 90%. These values prove the validity and accuracy of proposed technique.

For emphasizing on accuracy of proposed classifier, we compared it with RBF neural network. The confusion matrix of RBF-based classifier for radar clutters is shown in table 6. Although the results show that RBF-based classifier is also good but MLP-based classifier is much better than it and has a higher accuracy.

Table 6. confusion matrix of proposed RBF-based classifier

		Rayleigh	Log normal	weibull	K dist.
RBF	**Rayleigh**	**93.5%**	1.1%	0	5.4%
	Log normal	2.4%	**89.3%**	0	8.3%
	weibull	0	0.6%	**99.1%**	0.3%
	K dist.	1.4%	2.2%	4.9%	**91.5%**

In table 7, the comparison of results of proposed MLP-based method with different training algorithms and hidden layers with the results of RBF-based classifier are mentioned. The values of this table represent that proposed technique is more efficient.

Table 7. comparison of the accuracy of proposed MLP-based classifier and RFB-based classifier

Hidden layers	Training algorithm	% Accuracy of classifier
20* 15	**LM**	96.15%
30*15		95.65%
25		93.05%
35		95.75%
20* 15	**RP**	93.325%
30*15		94.55%
25		94.925%
35		95.65%
20* 15	**BFGS**	92.725%
30*15		96.575%
25		96.375%
35		95.35%
20* 15	**OSS**	95.9%
30*15		97.275%
25		96.75%
35		95.825%
20* 15	**SCG**	95.525%
30*15		95.6%
25		95.7%
35		96.225%
RFB		93.35%

6. Conclusions

Classification of radar clutters is very essential issue in radar researches. So in this paper is tried to classify four important models of clutter. Rayleigh, Log normal, Weibull and K distribution are these four models. Since to decrease the complexity of the classifier, a feature extraction has been designed for providing the classifier inputs. The proposed MLP-based classifier could classify clutters successfully. The results show that more than 90%, the proposed classifier was successful. The results of proposed technique were compared with the results of RBF-based method. All results prove the validity of proposed method.

Future researches will be focused on radar clutter classification with other kinds of Neural Networks and Soft computing algorithms. We will improve the Neural Networks by using soft computing algorithms like Genetic and Ant Algorithms. In the next works we will use some other features which can describe clutters better.

References

[1] J. Anderson, M. T. Gately, P. A. Penz, D. R. Collins, and others. "Radar signal categorization using a neural network," Proceedings of the IEEE, 1990 ,vol. 78, no. 10, pp. 1646–1657.

[2] J.-H.Lee, I.-S.Choi, H.-T.Kim. "Natural frequency-based neural network approach to radar target recognition," IEEE Trans. Signal Process, vol. 51, no. 12, p. 3191, December 2003.

[3] R. Rouveure, P. Faure, and Monod, "Multi-layer feed-forward perceptron for microwave signals processing," in Geoscience and Remote Sensing, Symposium. IGARSS '03, 2003, vol. 6, pp. 3519-3521.

[4] O. L. Mangasarian and W. H. Wolberg. Cancer diagnosis via linear programming. University of Wisconsin-Madison: Computer Sciences Department, 1990.

[5] T. L. Fine, Feedforward neural network methodology. Springer Science & Business Media, 2006.

[6] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," . The bulletin of mathematical biophysics, vol. 5, no. 4, pp. 115–133, 1943.

[7] S. Mirjalili, S. Z. M. Hashim, and H. M. Sardroudi. "Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm," Applied Mathematics and Computation, vol. 218, no. 22, pp. 11125–11137, 2012.

[8] S. Mirjalili and S. Z. M. Hashim. "A new hybrid PSOGSA algorithm for function optimization," in Computer and information application (ICCIA), 2010 international conference on, 2010, pp. 374–377.

[9] Z. X. Guo, W. K. Wong, and M. Li, "Sparsely connected neural network-based time series forecasting," Information Sciences, vol. 193, pp. 54–71, 2012.

[10] P. Auer, H. Burgsteiner, and W. Maass, "A learning rule for very simple universal approximators consisting of a single layer of perceptrons," Neural Networks, vol. 21, no. 5, pp. 786–795, 2008.

[11] Haykin and C. Deng, "Classification of radar clutter using NN," IEEE Trans. Neural Netw., vol. 2, November 1991.

[12] R. Soleti, L. Cantini, F. Berizzi, A. Capria, and D. Calugi, "Neural Network for polarimetric radar target classification," in Signal Processing Conference, 14th European, 2006, pp. 1–5.

[13] K. Cheikh and F. Soltani, "Application of neural networks to radar signal detection in k-distributed clutter," Radar, Sonar and Navigation, IEE Proceedings, vol. 153, no. 5, pp. 460-466, 2006.

[14] R. Vicen-Bueno, M. Rosa-Zurera, M. P. Jarabo-Amores, and R. Gil-Pita, "Automatic target detection in simulated ground clutter (Weibull distributed) by multilayer perceptrons in a low-resolution coherent radar," Radar, Sonar & Navigation, IET, vol. 4, no. 2, pp. 315–328, 2010.

[15] C. Bouvier, L. Martinet, G. Favier, H. Sedano, and M. Artaud, "Radar clutter classification using autoregressive modelling, K-distribution and neural network," in Acoustics, Speech, and Signal Processing, ICASSP-95., International Conference on, 1995, vol. 3, pp. 1820–1823.

[16] L. Pierucci and L. Bocchi, "Improvements of radar clutter classification in air traffic control environment," in Signal Processing and Information Technology, IEEE International Symposium on, 2007, pp. 721–724.

[17] L. Teng, H. Dan, "Model for spatial correlated clutter and its application to temporal spatialcorrelated clutter", IET Microwaves, Antennas & Propagation, Vol. 5 ,No. 3, pp. 298-304, 2011.

[18] K. D. Ward, S. Watts, and R. J. Tough, "Sea clutter: scattering, the K distribution and radar performance", IET, vol. 20, 2006.

[19] A. Farina, A. Russo, and F. A. Studer, "Coherent radar detection in log-normal clutter," Communications, Radar and Signal Processing, IEE Proceedings F,1986, vol. 133, no. 1, pp. 39–53.

[20] W. Stewing. Parametric spectral analysis of radar clutter. McMaster University, 1983.

[21] J. P. Burg, "A new analysis technique for time series data," presented at the NATO Advanced Study Institute on Signal Processing with Emphasis on Underwater Acoustics, Enschede, The Netherlands, 1968.

[22] S. S. Haykin, Adaptive filter theory. Pearson Education India, 2008.

[23] J. R. Barry, B. K. Carter, R. J. Erdahl, R. L. Harris, J. T. Miller, G. D. Smith, and R. M. Barnes, "Angel clutter and the ASR air traffic control radar," Applied Physics Laboratory, John Hopkins University, Silver Spring, MD, Final Report under Federal Aviation Administration Contract DOT-FA72WA-2705, 1973.

[24] M. Kubat, Neural networks: a comprehensive foundation by Simon Haykin, Macmillan: Cambridge Univ Press, 1999.

[25] H. Adeli and S.-L. Hung, Machine learning: neural networks, genetic algorithms and fuzzy systems. John Wiley & Sons, Inc., 1994.

[26] J. J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory," in Numerical analysis, Springer, pp. 105–116, 1978.

[27] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," Neural Networks, IEEE Transactions on, vol. 5, no. 6, pp. 989–993, 1994.

[28] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," The computer journal, vol. 7, no. 2, pp. 149–154, 1964.

[29] A. Ebrahimzadeh, A. Khazaee, "An efficient technique for classification of electrocardiogram signals, Advances in Electrical and Computer Engineering", Volume 9, pp. 89-93, 2009.
[30] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," in Neural Networks, IEEE International Conference on, 1993, pp. 586–591.
[31] S. Haykin, Neural Networks: A Comprehensive Foundation. New York: MacMillan, 1999.

Mohammad Akhondi Darzikolaei received the B.Sc and M.Sc degree from Babol Noshirvani University of technology. He worked on radar clutters and radar signal processing. He is now Telecommunication Ph.D student in Babol Noshirvani University of technology and his researches focus on radar and sonar signal processing, speech processing and pattern recognition and artificial neural networks.

Ata Ebrahimzade received the Ph.D degree in electrical engineering from Ferdosi University, Mashhad, Iran in 2007.
He is currently Professor with the Faculty of Electrical and Computer Engineering, Babol University of Technology, Babol, Iran. He has authored or co-authored more than 70 papers in international journals and conferences. His current research interests include signal processing and artificial intelligence.
Dr. Ebrahimzade is a reviewer of international conferences and journals.

Elahe Gholami received the B.Sc and M.Sc degree from Babol Noshirvani University of technology. She worked on cognitive radio spectrum sensing. Her researches focus on WSN, Cognitive radio, Signal processing, artificial neural networks and soft computing algorithms.