

Investigation of two Neural Network Methods in an Automatic Mapping Exercise

Sridhar Dutta¹, Rajive Ganguli² and Biswajit Samanta³

¹ *Graduate Student, Dept. of Mining Engineering, UAF, USA*

² *Associate Professor, Dept. of Mining Engineering, UAF, USA*

³ *Visiting faculty, Dept. of Mining Engineering, IIT, Kharagpur, India*

Email: ftsd1@uaf.edu **Email:** ffrg@uaf.edu **Email:** samantab@mining.iitkgp.ernet.in

Abstract: This paper investigates the performance of two neural network (NN) methods viz. a radial basis function network (RBFN) and a multilayer feed forward network (MFFN) to predict the radioactivity levels at a given test site. A comparative evaluation of the two networks is done using Root mean square error (RMSE), Pearson's Rsq, Mean error (ME) and Mean Absolute error (MAE). It was found that the RBFN performed marginally better compared to the other method

Keywords: Radial basis function, Feedforward network, Genetic algorithm, Wardnet Architecture.

1. INTRODUCTION

Reliable estimation of the natural radioactivity level has always been a challenging and intriguing task. Its detection is critical to preserve the state of the natural environment, and reduce health hazards. Particularly, radioactive components spread over their surrounding areas from the source of generation. As a result, there exists a complex pattern in their spatial distribution. However, if this spatial pattern is captured through some pattern recognition techniques, the radioactive level at a particular region can be reliably predicted. Usually, an estimate always has an uncertainty associated with it which constantly propels us for the search of a more reliable and robust estimation technique. Different estimation techniques can work under fundamentally different concepts. Recently, the success of artificial neural networks (ANN) as an estimation method has provided a new avenue for obtaining improved estimates (Yama, et al, 1999; Samanta, et al, 2003; Dutta, et al, 2005; Wu and Zhou, 1993; Polishuk and Kanevski, 2000). Among the various ANN alternatives, MFFN has been quite popular due to its efficacy in identifying the complex non-linear relationships that often exists in the input-output patterns thereby providing a better solution for capturing the difficult spatial patterns of the data. Despite its effectiveness, the MFFN can suffer from an extensive computational time requirement especially in the training phase. In this study, another class of ANN namely the RBFN has been investigated and its results compared with a popular MFFN. The biggest advantage of the RBFN over the MFFN lies in its simplicity and reduced computational time during training without losing the power of non-linear spatial mapping capability. However, selection of RBFN parameters prior to training can take time.

This paper investigates the applicability of the RBFN towards the detection of the natural radioactivity level along with various issues involved in constructing such model. The performance of the RBFN is compared with a conventional MFFN.

2. DESCRIPTION OF THE TASK

The exercise involved the detection of radioactivity levels at 808 monitoring stations from a total of 1008 monitoring stations at a given test site. Prior information was available in the form of first 10 days of measurements (i.e. 10 datasets) at 200 of the remaining stations.

The data was available in the form of X-coordinate (in m), Y-coordinate (in m) and mean gamma dose rate (nanoSieverts/hr). Two sets of data were released for the purpose of estimation at the 808 stations on the 11th day. The first dataset comprised of the measurements (X, Y and mean gamma dose rate) at the same 200 stations and only the X, Y location for the remaining 808 stations. The second dataset (Joker dataset), however, was a simulated dataset identical to the first dataset that was distributed, except for an accidental release of high level of radioactivity in the south-western corner of the region. Detailed description of the datasets can be found in Dubious and Galmarini, 2005.

3. METHODOLOGY

In this paper two different methods, namely, the RBFN and a MFFN have been explored for the purpose of radioactivity level detection. Two different techniques viz. the random method and genetic algorithm (GA) were considered to divide the data in order to train the NN models. In other words, the various models were built using four different approaches. In the first, a RBF model was trained on the randomly divided data and its parameters (centres and width) obtained in an unsupervised fashion. In the second approach, model was similar to the first except it was trained on the GA divided datasets. For the third, the RBF model was trained on the GA divided datasets, and determination of the model parameters was done using a supervised training pattern. The fourth model was a MFFN with a Wardnet¹ architecture employing the Levenberg-Marquardt (LM) training algorithm.

3.1 Functionality of Radial Basis Function (RBF):

The RBFN consists of a number of basis functions which are linearly weighted to produce an output. In these networks, the input variables undergo a nonlinear transformation at the hidden layer by the basis functions. The nature and the magnitude of the output emerging from the basis functions usually depends upon two factors (i) the type of basis function used, and (ii) the relative distance between the input data space to radial basis function centres. Broadly, there are three classes of basis functions available viz, gaussian, multi quadratic and thin plate spline. Among them gaussian functions are most commonly used. These functions are further characterized by the basis centres and a width or scale parameter σ . It is the number and type of the basis functions along with the various parameter values used in RBFN that gives it its corresponding feature detection capability. For example, a gaussian radial function is expressed as:

$$\Phi(\|x - \mu_c\|, \sigma) = \exp\left[-\frac{(\|x - \mu_c\|)^2}{\sigma^2}\right] \quad (1)$$

The final output, $y(x)$, is the weighted linear combination of the outputs from radial basis functions (Rao and Zhang, 2000)

$$y(x) = \sum_{i=1}^M w_i \phi(\|x - \mu_i\|) \quad (2)$$

where,

w = weights of the output layer

M = number of basis functions

¹ NeuroShell 2, Release 4, Copyright 1993-2000 by Ward Systems Group[®]

Figure 1 shows a schematic diagram of a radial basis function network where the input variable X is mapped in to the output Y . Since the output is a linear function of unknown weights w_i , they can be computed easily by simply a matrix inversion. Hence, it largely reduces the computation time during training phase.

Figure 2 is a simplified illustration on the working principle of radial basis function. In the example: A, B, C and D are four basis functions with fixed centre in the X-Y plane. The width of the basis function determines its range of influence. While detecting the radioactivity level at an unknown point, the degree of influence of a particular basis function depends on the proximity of its centre to the point under consideration such that if the point falls outside the its range of influence it wont have any effect on the output value of the radioactivity level. For example, for the point marked 1, only the outputs of radial basis functions A and B will be linearly combined to compute the radioactivity level. Moreover, since it is closer to the centre of basis function A than B, it will be influenced more by A. Similarly, the radioactivity level at the point 2, is influenced by all the four basis functions, while, point 3, is impacted only by the basis function B.

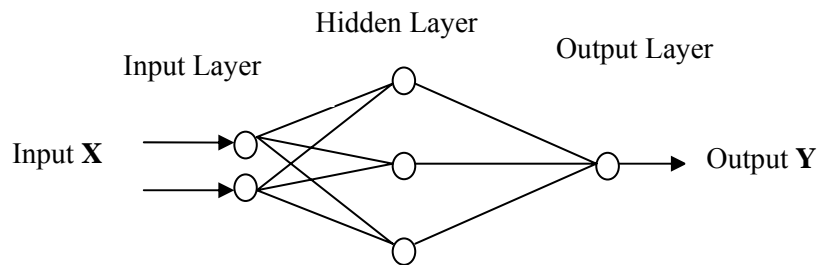


Figure 1: Schematic diagram of RBFN

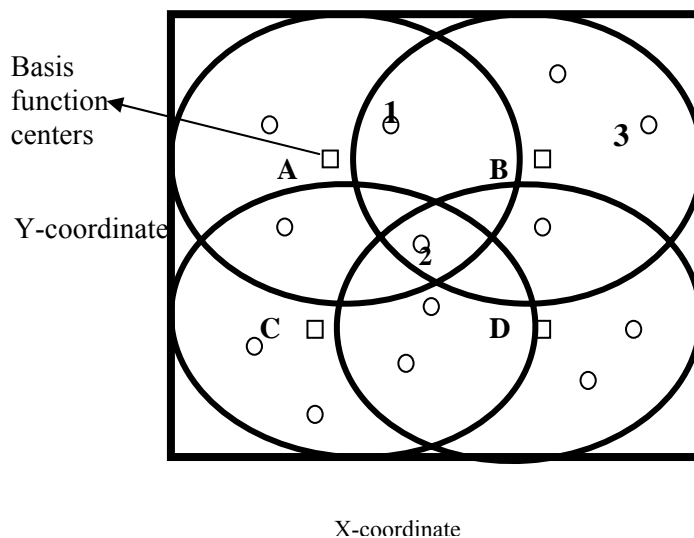


Figure 2: Working principle of Radial basis functions

3.2 RBF Modeling:

The primary goal in any modelling task is to find a fit to the statistical process responsible for the data. While doing RBF modelling the number and the type of the basis function along with its parameters decide the overall performance of the network. The number of basis functions is vital as too high or too little of it can lead to the conditions termed as "over-fitting" or "under-fitting". While an over-fitted model memorizes the noise in the data an under-fitted model fails to capture relevant features existing in the data. Similarly, as mentioned in the previous section the width/scale parameter decides the degree of influence of a particular basis function. Apart from that, optimal selection of basis centres is another critical element. These issues, therefore, require a sensible selection of the model parameters for its proper generalization, i.e. so that it captures the most regularized variation of the system without fitting to noise in the data. There are several approaches (Haykins, 1999; Orr, 1996; Howlett and Jain, 2001) for the selection of the centre vectors. These may include the random method of selection, selection in a supervised fashion employing the standard gradient descent approach, or through the various clustering algorithms such as the Kohonen network or the self-organizing maps. Among these, the random selection technique is the simplest in which M out of the N training data points are randomly selected. This is a trial and error approach, such that the M chosen points is used to calculate the RMSE on an independent calibration dataset. The optimum set of the centres points is the one that minimizes this error. Once the centre vectors are decided, the next step involves an appropriate choice of the width/scale parameter for the basis function as it affects the degree of smoothness associated with the output function. The width parameter can either be fixed to a particular value for all the basis functions or can be varied. In general, a common width is set to all the basis functions, which is some multiple of the average distance between the basis centres. Finally, the weights of the output layer are determined as the least square solution to the minimization of the total sum squared error. According to this the weight matrix for the output layer is given by:

$$W = (G^T * G)^{-1} * G^T * Y \quad (3)$$

where,

$$G = \begin{vmatrix} \phi(\|x_1 - \mu_1\|) & \dots & \phi(\|x_1 - \mu_M\|) \\ \phi(\|x_N - \mu_1\|) & \dots & \phi(\|x_N - \mu_M\|) \end{vmatrix}$$

$$Y = (y_1, y_2, \dots, y_{N-1}, y_N)$$

The adverted method of selecting the basis centres and the width randomly, neglecting the output patterns in an unsupervised manner, may at times lead to the sub-optimal choice of these parameters. Therefore, supervised selection of the network parameters i.e. the centres, width and output weights were done employing the standard gradient descent technique. In this approach the error function 'E' given by equation 4 is minimized with respect to each parameter of the model to get the updated values of the parameters for the subsequent iterations. For the gaussian radial basis function, the updated equations for the weight, width and the center locations are given by equation 5, 6 and 7.

$$E = \sum_k (t_k^n - y_k(x^n))^2 \quad (4)$$

$$\Delta w_{kj} = \eta(t_k^n - y(x^n))\phi_j(x^n) \quad (5)$$

$$\Delta \mu_j = \eta \phi_j(x^n) \frac{\|x^n - \mu_j\|}{\sigma_j^2} \sum_k \{y_k(t_k^n) - x^n\} w_{kj} \quad (6)$$

$$\Delta \sigma_j = \eta \phi_j(x^n) \frac{\|x^n - \mu_j\|^2}{\sigma_j^3} \sum_k \{y_k(t_k^n) - x^n\} w_{kj} \quad (7)$$

where, t_k^n is the target value for the output unit k when the network is presented with the input vector x^n . Though the supervised selection procedure may lead to the optimal choice of the parameters, it has disadvantages when compared to the unsupervised method stated earlier. First, it is computationally intensive and second, there is no guarantee that with this kind of training the basis functions will remain localized and if it is the case, it may lead to inferior network performance (Nag and Ghosh, 2001). Both these procedures have been looked upon in this paper. The RBFN modelling was done using self developed MATLAB code.

3.3 Multilayer Feedforward Network Modeling:

A MFFN with a ward-net architecture, as shown in Figure 3, was chosen for the NN modeling. The network comprised of 5 slabs: one input slab, 3 hidden slabs and 1 output slab (slab is basically a group of neurons; a particular layer may have multiple slabs). Each slab in the hidden layer and the output layer consisted of different activation functions. The input slab has two neurons for each of the X and Y coordinates while the output slab has one neuron for the radioactivity level. The three slabs in the hidden layer used three different activation functions viz. tanh, gaussian and complementary gaussian. The idea behind using different combinations of the activation functions was to identify various dissimilar portions of the dataset as a particular activation function may act good for few typical patterns and may not work at all for others. Thus, using different activation functions ensures that at least some of the underlying trends in the data are captured. For example, a gaussian activation function in one hidden slab may detect features in the mid-range of the data while a gaussian complement activation function in another hidden slab may detect features from the upper and lower extremes of the data. Similarly, a tanh activation function will tend to squeeze together data at the low and high ends of the original data range which may be helpful in reducing the effects of outliers. Thus, the network gets different views of the data. Combining these three features in the output layer is expected to produce better predictions. For the convenience of the readers, the characteristic equation of the output signal from the gaussian, complementary gaussian and tanh activation function are given by equations 8-10.

$$\text{Gaussian activation function} = \exp(-x^2) \quad (8)$$

$$\text{Gaussian-complement activation function} = 1 - \exp(-x^2) \quad (9)$$

$$\text{Tanh activation function} = \frac{\exp^x - \exp^{-x}}{\exp^x + \exp^{-x}} \quad (10)$$

where, x is the input to the activation function. A MATLAB program applying LM algorithm with the batch mode method of selection was developed and used for the training purpose. Detail description on the LM algorithm can be found at Hagan et al, 1995.

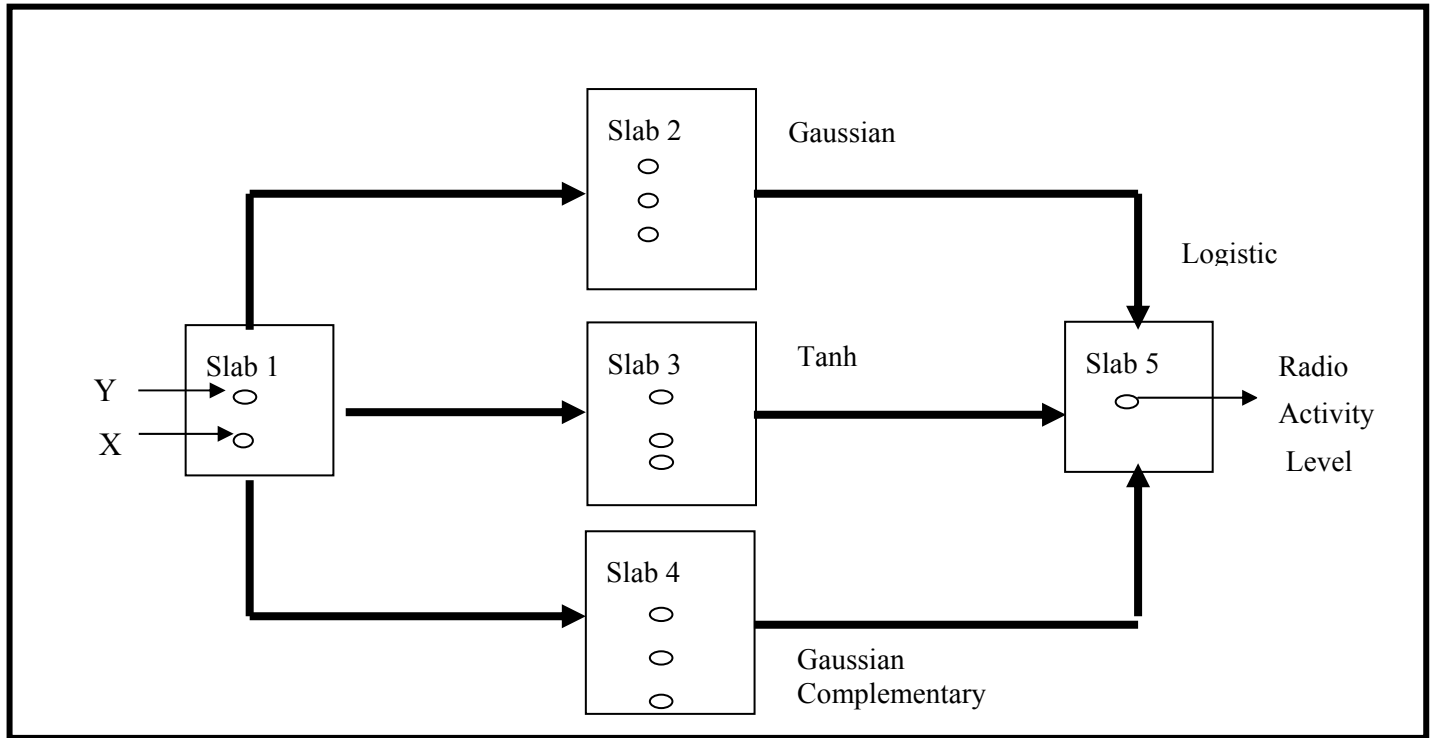


Figure 3: Ward Net Architecture

3.4 Model Development:

The main goal of NN modelling, among several other related issues, is to ensure proper model generalization. Neural networks, both RBFN and MFFN, are very flexible models; and hence besides fitting to the regular complex curvilinear patterns of the process, they may also get fitted to the irregular noisy component in the data, if they exist. Therefore, proper care should be taken to restrain the network from fitting to the noise in the data. Thus, to ensure model generalization an optimal network has to be chosen by some means. In RBFN, this pertains to the selection of the optimal number of basis functions and their basis centres and widths. In MFFN, although selection of the number of hidden neurons is critical; the model, however, could be generalized with arbitrarily high number of hidden neurons by ceasing the training process at the right time using early-stop method. Note that the early-stop training, which was employed in this study, is one of the several popular methods used for the MFFN model generalization (Haykins, 1999). In order to select an optimal generalized network (either RBFN or MFFN), the network is trained using the training samples, however, its performance is observed on a calibration dataset, independent of the training dataset. The optimal network is selected which produces least error on the calibration dataset. Thus, following the above exercise the network model is calibrated properly.

It is noteworthy to point out that for a legitimate calibration exercise, the properties of the training and calibration datasets should be statistically similar. Otherwise, network will

not produce the desired generalized performance. Unfortunately, the common practice is to, randomly subdivide the available samples at the model development stage into training and calibration subsets. However, this random selection of samples produces several undesirable characteristics including statistical dissimilarity among the data sets, especially when the data are few and sparse (Bowden, *et al*, 2002). In this regard, Ganguli and Bandopadhyay, 2003a; Ganguli, *et al*, 2003b; Samanta *et al*, 2004 explored the possibility of using GA for the data division in neural network modelling. They achieved good success while applying GA for generating statistical similar datasets.

A simple way to estimate the generalization performance of the network is to measure the error it makes on a separate dataset (prediction set) which is unseen during the training process. Thus, the entire dataset can be divided into three subsets i.e. training, calibration and the prediction dataset. However, in this exercise the prediction dataset (808 observations) was unknown. So the known data (200 points) was divided into two subsets viz. the training subset and calibration subset using two different procedures in 4:1 proportion. In one case, the training data (160) and the calibration data (40) were obtained randomly while in another case it was obtained using GA. The use of GA ensured that the datasets were statistically similar, since there is no guarantee of them being similar if obtained randomly. Data division using GA has been described in the next section. Table 1 and Table 2 show the statistics for the 1st and 2nd dataset respectively. Initially, for the first dataset, the 200 observations from the first ten days of measurement and from the 11th day were averaged to get a single dataset of 200 observations. However, for the second dataset (joker dataset) this prior information available from the first 10 days of measurement was not considered as it contained few high outlier values in the dataset released on the 11th day. It was seen that with RBF modelling 16 centres and for MFFN modelling 9 hidden neurons produced the optimum results. The performance was validated by the mean square error (MSE) and Rsq values in the validation dataset of 40 observations. This analysis was performed for several network architectures by varying the number of hidden neurons, number of hidden layers, type of activation function in case of MFFN modelling and number of centres in case of RBF modelling. Further, in RBF modelling supervised selection of the model parameters described in the earlier section were also carried out using the GA divided datasets. After the models were developed they were tested on the 800 test data points.

GA division 1 st dataset	X	Y	Radioactivity	X	Y	Radioactivity
	Training Set			Calibration Set		
Mean	8.5 e04	3 e05	96.27	8.5e04	3e05	96.01
Std Dev.	9.8e04	2.1e05	17.67	9.6e04	2.1e05	17.74
Random Div 1 st Dataset						
Mean	8.4e04	3e05	95.8	9.0e04	2.9e05	97.21
Std Dev.	9.5e04	2.2e05	16.23	1.1e05	1.8e05	22.65

Table 1. Statistics of data-division for the 1st dataset.

GA division 2 nd dataset	X	Y	Radioactivity	X	Y	Radioactivity
	Training Set			Calibration Set		
Mean	8.5e04	2.9e5	107	8.3e04	3e05	117.6
Std Dev.	9.6e04	2.1e05	109.5	1.0e+05	2.1e05	154.9
Random Div 2 nd Dataset						
Mean	7.6e04	2.9e05	110	1.2e 05	3.2e05	99.2
Std Dev.	9.8e04	2.1e05	134.2	8.7e04	2.0e05	25.7

Table 2. Statistics of data-division for the 2nd dataset.

3.5 Data Division using Genetic Algorithm:

Genetic algorithms (GA) has been popularly used in many scientific domains such as computer science (Bhuyan, 1995), operation research (Fang, *et al*, 1993), social science (Holland, 1975), biology (Kremer, 1992), mining engineering (Ganguli, *et al*, 2003b). Recently, it has been widely applied for data divisional purpose (Bowden, *et al*, 2002; Ganguli & Bandopadhyay, 2003a; Ganguli, *et al*, 2003b; Samanta, *et al*, 2004). However, these researchers used the GA to split the entire dataset into three subsets i.e. the training, the validation and the prediction subset.

GA is an optimization technique based on the theory of genetics and natural selection (Goldberg, 1999). It divides the data in such a way that the statistical difference is minimized for the subsets of the dataset. The reason for division into subsets has been mentioned in the previous section.

The data division basically consists of the following cycle of steps:

- Evaluate the fitness of all individual data division in the population. In the present study each individual represents a random replicate of the original dataset.
- Create a new population by performing operations such as crossover and mutation on the individuals based on a fitness function.
- Discard the unfit population, generate a new population and iterate using the new population as shown in Figure 1.

A generation is basically one cycle of the above three steps to form a possible solution. The first generation (Generation = 0) operates on a population of randomly generated individuals. From thereon, genetic operation, improves the population. GA used for data division sorts the samples into the subsets by using a set of random numbers. To accomplish this, a random number seed is generated. This random number seed controls the generation of random sequence of numbers from which 'n' population members of the individual datasets are created. Pairs of members are selected and the genetic operations of cross over and mutation are applied to obtain a new improved population. To evaluate the "fitness" of each solution an objective function is required, which would minimize the sum of the modified t-stat between each pair of the three subsets as specified by (Ganguli, 2003b). The modified t-stat function is given by,

$$\text{modified t-stat} = \frac{|\text{mean1} - \text{mean2}|}{\sqrt{\frac{\text{var1}}{n1} + \frac{\text{var2}}{n2}}} + \frac{|\text{var1} - \text{var2}|}{\left(\frac{\text{var1} + \text{var2}}{2}\right)} \quad (11)$$

where, mean1, var1, n1 are the mean, variance and number of elements in group1, while mean2, var2, n2 are the mean, variance and number of elements in group2. Note that the modified t-stat sets the definition of equivalency. This definition should be changed to suit the problem. Simply speaking, minimization of Equation (11) constitutes minimization of the difference between the means and variances of subsets, i.e. the subsets would have similar means and variances. For non-normal data, this may not be the right approach to equivalency. For non-normal data, one could try ensuring similarity using simple approaches such as difference between various deciles (such as 10th decile, 30th decile and median).

4. RESULTS AND DISCUSSION

The statistical properties of the training, calibration and predictions are presented in Tables 3 and 4. It should be noted that the prediction subsets were not known. Tables 5 through 9 present the statistical summary of the estimates and their respective errors for the two test datasets used in this study. Figures 4 through 11 show the contour plots of the estimated radioactive levels at the region using the various methods while Figures 12 through 15 present the 3-D surface plots of the estimated values for the 2nd dataset which gives another view of the radioactivity level distribution at the region. It could be seen from Table 6 and Table 8 that for the 1st dataset almost all the methods performed equally well with the exception of the RBF method with GA data division and supervised training. This may be due to the basis functions not staying localized. However, the same RBF method with the GA data division and unsupervised training had a slight edge over the other two in terms of Pearson's r and RMSE. There was not much difference in the results for these three methods, which may be due to the very similar properties of the training and calibration dataset obtained from the GA and random data-division (Table 3) on which the model was trained. The MFFN, however, had a large bias. It is also revealed that the neural network models work appreciably better for the first data set in comparison with the 2nd data set. This result is quite expected since the 2nd data set contained some outlier values. It is further revealed from Figures 7, 9 and 11 that although the methods MFFN and RBF with GA data division are able to detect the anomalies that exist in the south-western corner of the second dataset, they are not able to detect the large magnitudes of the outliers. The MFFN model in this case also had a large bias and RMSE. From Table 6 and Table 9 it could be seen that in case of the 2nd dataset the RBF model with random data division didn't work well compared to the RBF model with GA data division. This can be attributed to the very different training and calibration sets obtained from GA and random data division (Table 4). It could be seen from Table 4 that, the range of the radioactivity value in the calibration set was far less than that of the training set. So the model might have been under trained, when it predicted the 808 points on the prediction dataset. However, this was not the case with the GA data-division for which the training and the calibration subsets had almost similar properties. So the magnitude of the predictions was also large for the GA division model compared to the random division model.

As far as the uncertainties in prediction are concerned, the standard deviation can be considered a crude yardstick of the measure of uncertainty associated with the prediction performance. Further, it is revealed that the execution time of RBFN network was less than a minute using a standard Pentium 4 machine. The MFFN took around 10-15 minutes to get executed. Therefore, the RBFN network might be preferable for large data set where execution time is also an important factor.

	Min	Max	Mean	Median	Std. Dev
Random Division					
Training	58.8	138	95.80	97.5	16.23
Calibration	58.2	153	97.21	98.1	22.65
Prediction	57	180	98.02	98.80	20.02
GA Division					
Training	58.2	153	96.27	97.8	17.67
Calibration	68.8	140	96.07	95.2	17.74
Prediction	57	180	98.02	98.80	20.02

Table 3. *Statistical properties of the first dataset (Radioactivity Levels)*

	Min	Max	Mean	Median	Std. Dev
Random Division					
Training	58.8	1499	110	97.5	134.2
Calibration	58.2	196.1	99.2	100.8	25.7
Prediction	57.00	1528.80	105.40	98.95	83.71
GA Division					
Training	58.2	1499	107	99.65	109.5
Calibration	58.8	1070.4	117.6	95.6	154.9
Prediction	57.00	1528.80	105.40	98.95	83.71

Table 4. *Statistical properties of the second dataset (Radioactivity Levels)*

N=808	Min	Max	Mean	Median	Std. Dev
True (First Dataset)	57.00	180.00	98.02	98.80	20.00
Estimates (First Dataset)	67.40	126.60	97.70	99.30	14.30
True (Second Dataset)	57.00	1528.80	105.40	98.95	83.71
Estimates (Second Dataset)	61.40	129.80	103.30	103.40	17.01

Table 5. Comparison of the estimated and measured values (nSv/h)
(Original Method: RBF+ random data division)

First Dataset	N = 808	Min.	Max.	mean	median	Std. Dev.
True Values (First Dataset)		57	180	98.0	98.8	20.0
RBF (Random Division)		67.4	126.6	97.7	99.3	14.3
RBF (GA division + Unsupervised training)		65.6	123.4	97.2	98.9	14.1
RBF (GA division + Supervised training)		63.8	108.93	96.5	98.8	10.1
LMA		68.2	126.8	100.2	101.2	14.60
Second Dataset	N = 808					
True Values (Second Dataset)		57	1528.8	105.4	98.95	83.71
RBF (Random Division)		61.4	129.8	103.3	103.4	17.01
RBF (GA division + Unsupervised training)		45.9	191.9	106.7	101.62	37.23
RBF (GA division + Supervised training)		50.1	203.8	105.2	101.63	38.84
LMA		72.5	273.9	123.4	103.5	51.4

Table 6. Comparison of the True and estimated values (nSv/h) (Other Methods)

Data sets:	MAE	ME	Pearson's r	RMSE
First data set	9.92	0.2	0.76	13.1
Second data set	17.5	5.1	0.29	80.6

Table 7. Comparison of the errors (Original Method, RBF random division)

Data sets:	MAE	ME	Pearson's r	RMSE
Original Method (Random)	9.92	0.2	0.76	13.1
RBF (GA division + Unsupervised Training)	9.62	0.90	0.78	12.7
RBF (GA division + Supervised training)	12.2	1.50	0.64	15.9
LMA (Ward Net)	9.93	2.18	0.76	13.3

Table 8. Comparison of the errors for 1st dataset (Other Methods)

Data sets:	MAE	ME	Pearson's r	RMSE
Original Method (Random)	17.5	5.10	0.29	80.6
RBF (GA division + Unsupervised Training)	28.2	-0.22	0.31	80.1
RBF (GA division + Supervised training)	28.9	-1.29	0.33	79.9
LMA (Ward Net)	38.5	17.98	0.27	87.3

Table 9. Comparison of the errors for 2nd dataset (Other Methods)

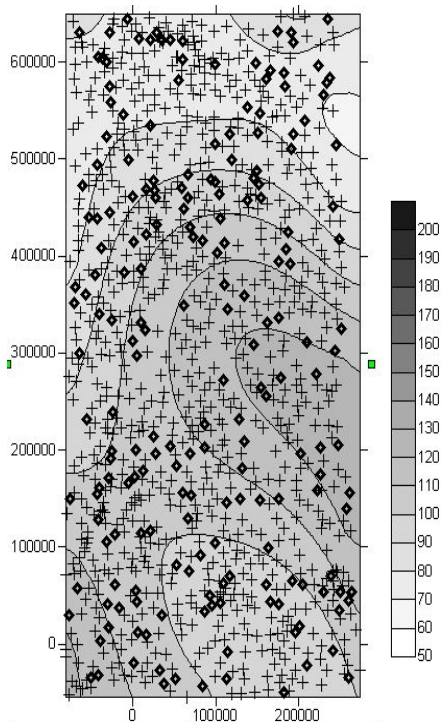


Figure 4: 1st Dataset Random RBF

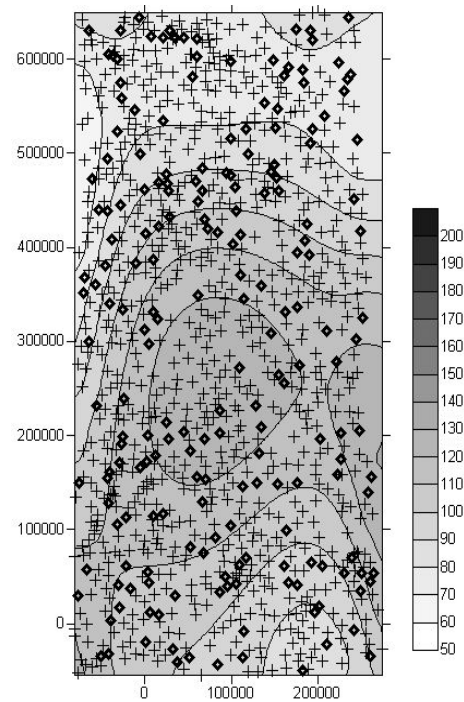


Figure 5: 2nd Dataset Random RBF

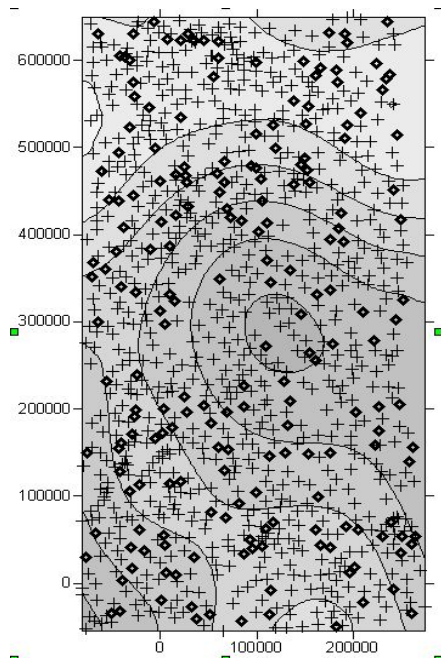


Figure 6: 1st Dataset GA division + Unsupervised Training RBF

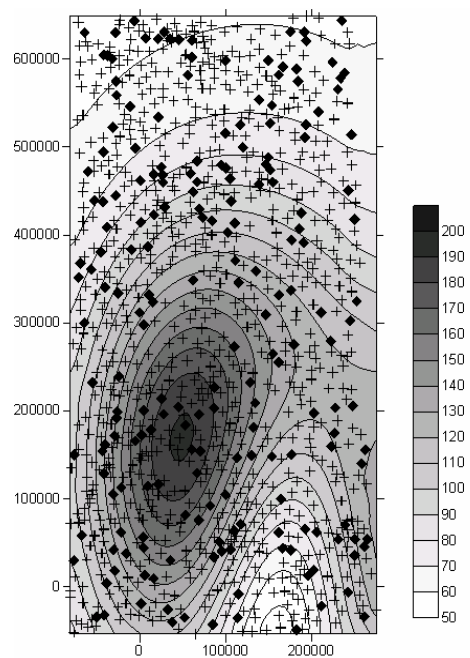


Figure 7: 2nd Dataset GA division + Unsupervised Training RBF

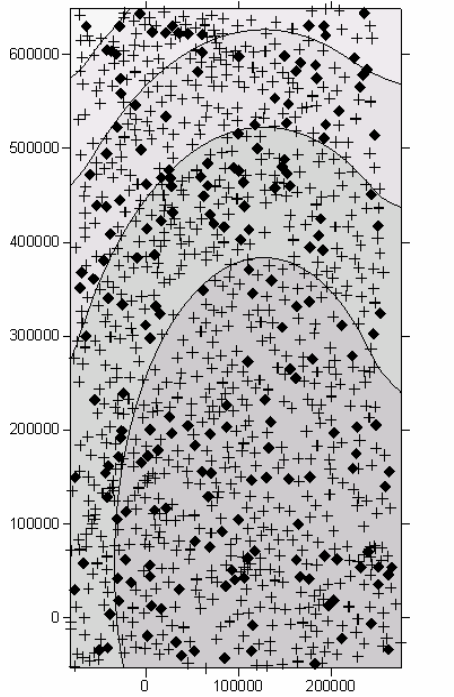


Figure 8: 1st Dataset GA division + Supervised Training RBF

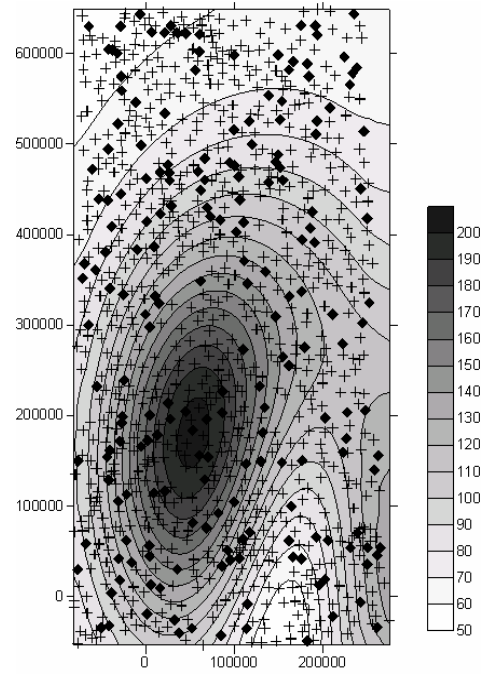


Figure 9: 2nd Dataset GA division + Supervised Training RBF

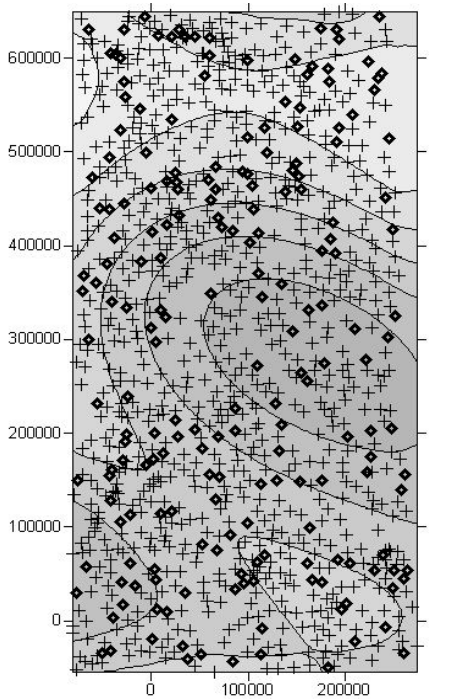


Figure 10: 1st Dataset feedforward

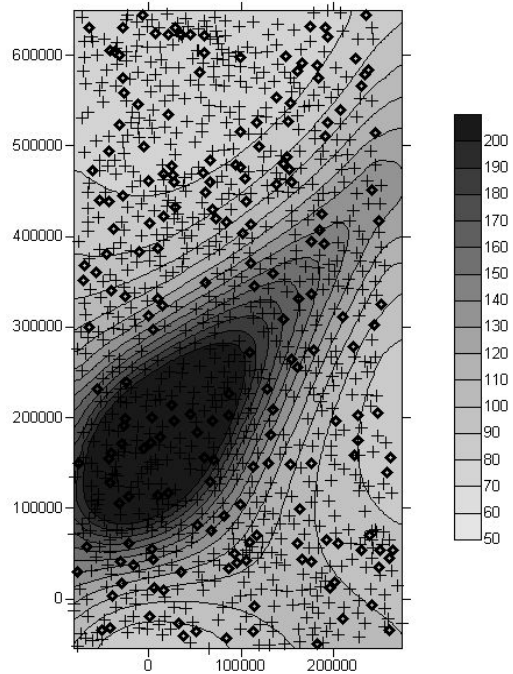


Figure 11: 2nd Dataset feedforward network

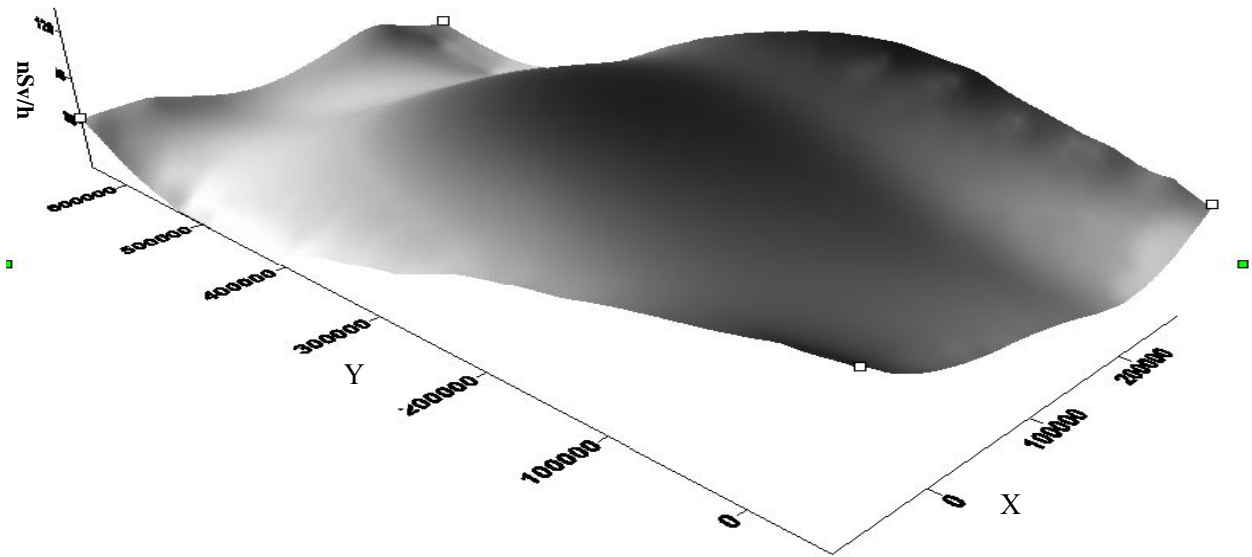


Figure 12: 3D Surface Plot for 2nd Dataset (Random Data Division)

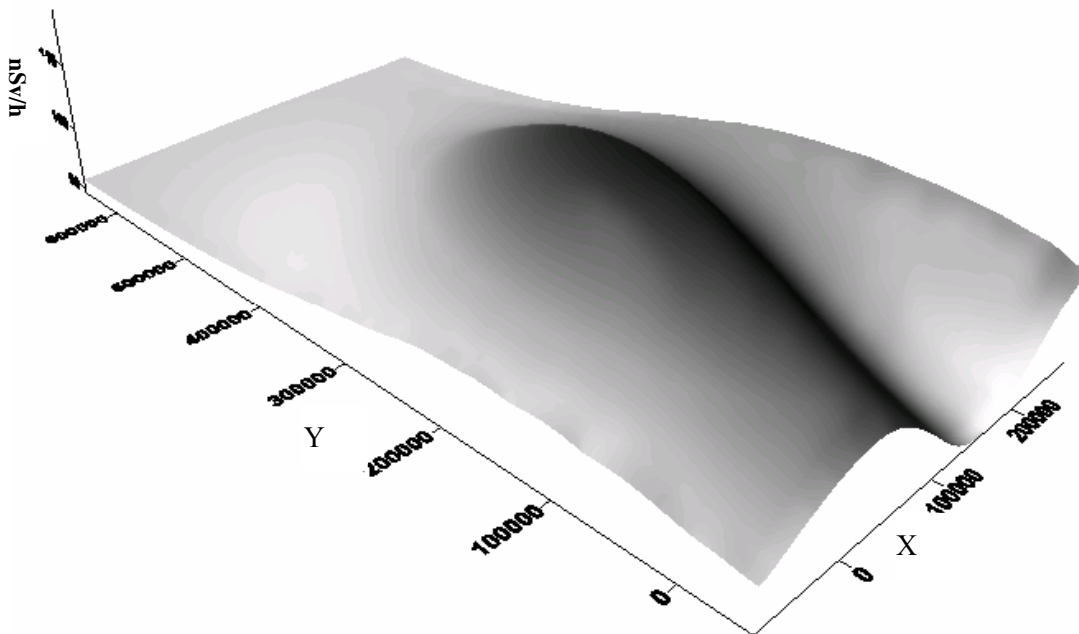


Figure 13: 3D Surface Plot for the 2nd Dataset (GA division + Unsupervised Training)

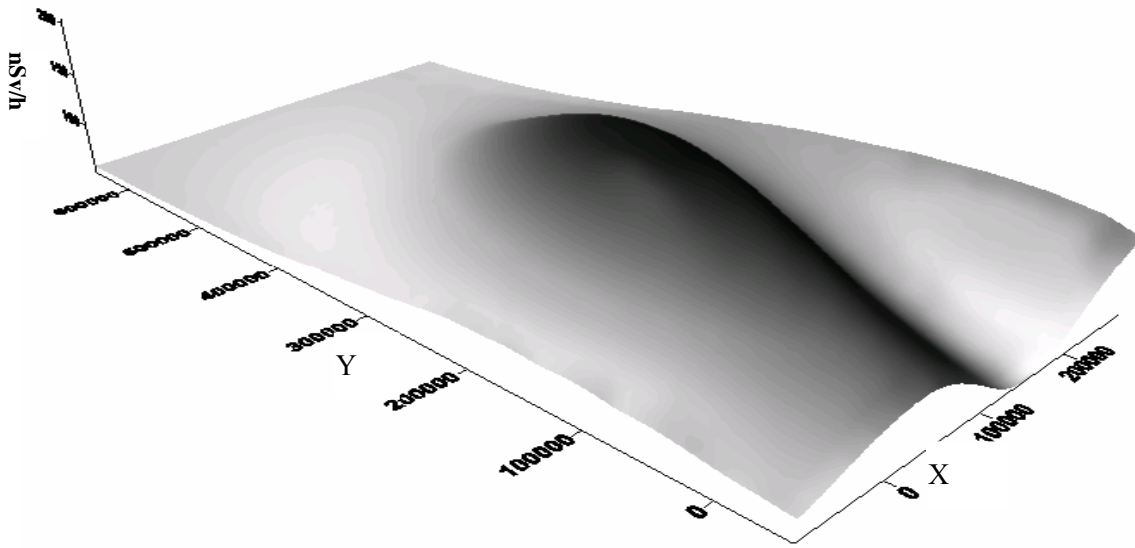


Figure 14: 3D Surface Plot for the 2nd Dataset (GA division + Supervised Training)

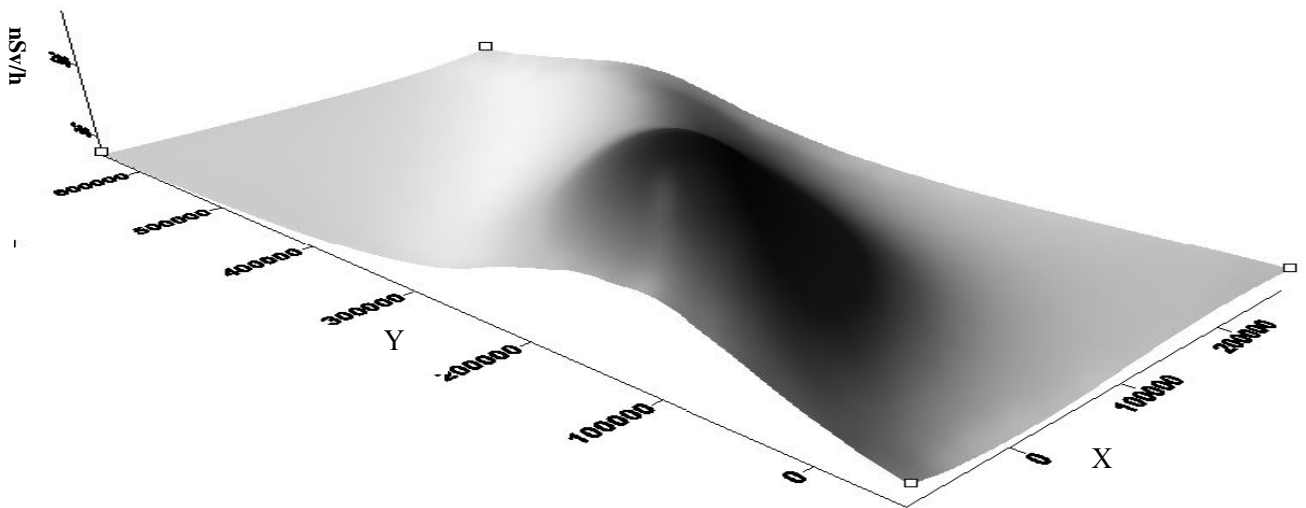


Figure 15: 3D Surface Plot for 2nd Dataset (Feed forward network)

5. CONCLUSION

In this paper two NN modelling techniques were investigated to estimate the radioactivity levels at the given test site: one using the RBFN and the other employing a MFFN. Usually, in NN modelling the three subsets (training, calibration and prediction) need to be statistically similar. The model is trained on the training subset and predictions are made on the prediction dataset hoping that the information available is representative of the real world situation. But this was not the case with us. The prediction dataset was unknown and so, it was of interest to see how the selection of training and calibration subset could affect the model performance. During the model development stage, the data was divided into two subsets: training and the calibration. The training data was used to train the NN models, and the calibration data was used to generalize (calibrate) the models. In one case the training and the validation data set were obtained by random division from the original set of 200 observations, while in the other case it was obtained using GA to ensure statistical similarity. The network parameters i.e. the number of centres for RBFN, and the number of hidden neurons for MFFN were determined by observing the performance of the models on the calibration dataset. Apart from this, for the RBF model built on the GA divided datasets, both supervised and unsupervised training approaches were explored. The actual performance of the models was tested on 800 data points supplied for the prediction purpose. From the results, it is evident that neither of the methods performed well in the 2nd dataset; although the performance was reasonable well for the 1st data set. The performance (in the prediction subsets) of the various models were probably governed by the similarities (or lack thereof) between the training/calibration subsets and the prediction subsets. For example, in the second dataset, the standard deviations of the training/calibration subsets are very different from that of the prediction subset. Additionally, due to relative sparseness of the training/calibration subsets, the two subsets (in both datasets) did not achieve the degree of similarity as one would desire. The performance of the RBF with GA data division and supervised training of gradient descent algorithm was marginally better. However, such an improvement might not be statistically significant. On the other hand, the gradient descent approach for the RBFN approach didn't improve the model performance for the 1st dataset.

Dutta, S., Ganguli, R. and Samanta, B., 2005, "Investigation of two Neural Network Methods in an Automatic Mapping Exercise," **Applied GIS**, Vol. 1 No. 2.

References:

Bhuyan, J. (1995) A combination of genetic algorithm and simulated evolution techniques for clustering Full text, *Proceedings of the 1995 ACM 23rd annual conference on Computer science*, Nashville, Tennessee, USA, pp127-134, ISBN:0-89791-737-5

Bowden, G. J., Maier, H. R. and Dandy, G.C. (2002). Optimal Division of Data for Neural Network Models in Water Resources Application, *Water Resources Research*, Vol. 38, No. 2, 11 pp.

Dubois, G., Galmarini, S. (2005). "Introduction to the Spatial Interpolation Comparison (SIC) 2004 exercise and presentation of the data sets". This Volume.

Dutta, S., Ganguli, R., & Samanta, B. (2005) Comparative Evaluation of Radial Basis Functions and Kriging for Ore Grade Estimation, *32nd International Symposium of the application of Computers and Operation research in Mineral Industry, APCOM 2005*, (Accepted).

Fang, H.L., Ross, P. and Corne, D. (1993) A Promising Genetic Algorithm Approach to Job-Shop Scheduling, Rescheduling, and Open-Shop Scheduling Problems *Proc. of the Fifth Int. Conf. on Genetic Algorithm*.

Ganguli, R. and Bandopadhyay, S. (2003a). Dealing with Sparse Data Issues in a Mineral Industry Neural Network Application, *Proceedings, Computer Applications in the Mineral Industry (CAMI)*, Calgary, Alberta, Canada, 5p.

Ganguli, R., Walsh, D.E. and Yu., S. (2003b). Calibration of On-line Analyzers Using Neural Networks, Final Report to U.S. Department of Energy, Contract #: DE-FC26-01NT41058.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine Learning*, Pearson Education Asia Pte Ltd. 412 pp.

Hagan, M.T., Demuth, H.B. and Beale, M. (1995). *Neural Network Design*, PWS Publishing Company, Boston, 19 chapters.

Haykins, S. (1999). *Neural networks: A comprehensive foundation (2nd Ed)*: Prentice Hall, New Jersey, 824 pp.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*, Ann Arbor: The University of Michigan press.

Howlett, J.R., Jain, C.J(2001). *Radial Basis function networks 2*, ISBN 3-7908-1368-0.

Kremer, S.S. (1992) Genetic Algorithms for Protein Tertiary Structure Prediction, *Parallel Problem Solving from Nature II*, Ed. R. Männer, B. Manderick North Holland, pp. 391-400.

***Matlab-The language of technical computing* The Mathworks Inc. copyright 1984-2002**

Nag, A. and Ghosh, J. (2001). An overview of radial basis function networks *Radial basis function networks 2* ed. Robert J. Howlett and Lakhmi C. Jain , ISSN-1434-9922.

***NeuroShell 2, Release 4*, Copyright 1993-2000 by Ward Systems Group[®]**

Orr, M. J. L. (1996). Introduction to radial basis function networks *Technical report*, Center of cognitive science, University of Edinburgh, April.

Polishuk, V. and Kanevski, M. (2000). Comparison of Unsupervised and Supervised Training of RBF Neural Networks. Case Study: Mapping of Contamination Data, *Proceedings of the Second ICSC Symposium on Neural Computation* May, Berlin, Germany, pp. 641-646.

Dutta, S., Ganguli, R. and Samanta, B., 2005, "Investigation of two Neural Network Methods in an Automatic Mapping Exercise," **Applied GIS**, Vol. 1 No. 2.

Rao S.G., and Zhang, B. (2000) Radial Basis function networks, *Artificial Neural Networks in Hydrology*,

Samanta, B., Ganguli, R., and Bandopadhyay, S. (2003), Comparing the Predictive Performance of Neural Networks With Ordinary Kriging in a Bauxite Deposit, *Transactions of Institute of Mining and Metallurgy, UK*, (Under review)

Samanta, B., Bandopadhyay, S., and Ganguli, R., (2004) Data Segmentation and Genetic Algorithms for Sparse Data Division in Nome Placer Gold Grade Estimation Using Neural Network and Geostatistics, *Exploration and Mining Geology*, vol.11, pp. 69-76.

Yama, B. R. And Lineberry, G. T. (1999) Artificial Neural Network Application For A Predictive Task In Mining, *Mining Engineering*, Feb,pp. 59-64.

Wu, X., and Zhou, Y. (1993) Reserve estimation using neural network techniques, *Computer & Geoscience*, Vol. 19, No. 4, pp. 567-575.