J. Statist. Res. Iran **16** (2019): 211–228
DOI: 10.52547/jsri.16.1.211

The Ability of Artificial Neural Networks in Learning Dependency of Spatial Data

A. Tavassoli[†], Y. Waghei[*,†] and A.R. Nazemi[‡]

[†] University of Birjand
[‡] Shahrood University of Technology

Received: 8/4/2020 Approved: 1/2/2021

Abstract. In conventional methods of spatial data analysis, such as Kriging, the dependency structure of data is estimated, modeled, and then used for data prediction. In contrast, the Artificial Neural Network (ANN) approach, which is a data-driven approach, does not model the data dependency structure. Therefore, an important question may arise here: Does ANN use, indirectly, spatial dependency structure in data prediction? In this paper, we want to answer this question through a simulation study. Different dependent and independent spatial data sets are simulated under two spatial structures, and the prediction accuracy of ANNs is compared for simulated data. It is shown that neural network error for predicting dependent spatial data is much less than that of independent spatial data. We conclude that the network can indirectly learn spatial dependence between the observations. We also applied the ANN method to an experimentally obtained data set and compared its prediction accuracy with Kriging as a common geostatistical method. The results showed that the neural network can be used as an alternative method for spatial data prediction.

Keywords. Artificial Neural Networks; Spatial dependency; Spatial prediction.
MSC 2010: 62M45, 62H11.

[*] Corresponding author

1 Introduction

The spatial prediction has already been investigated using methods such as Kriging, inverse distance weighting, interpolating polynomials, splines, etc., and different studies have compared the accuracy of these methods (Li and Heap, 2011; Zimmerman et al., 1999; Mojiri et al., 2018). Among these methods, Kriging, which directly accounts for the spatial dependency structure through the Variogram model, would be an appropriate option and has been widely applied for spatial data prediction (Cressie, 1993; Webster and Oliver, 2007; Gelfand et al., 2010; Carlin et al., 2014).

In recent years, artificial neural networks have broadly been used in Engineering, for pattern recognition and data classification, and in Sciences, especially in Statistics, for regression, time series, and even spatial analysis (Samarasinghe, 2016; Khashei-Siuki and Sarbazi, 2015). The neural network has strong nonlinear mapping power and self-learning capability to learn from data, even when its user does not have much information about the data model. In conventional methods of spatial data analysis, such as Kriging, the dependency structure of data is estimated, modeled, and then used in data prediction, under some conditions such as stationarity. These methods do not perform well without accurately modeling the spatial dependency structure between data, which needs expert knowledge. In contrast, the neural network approach is data-driven (not model-driven) and does not model data dependency form; Still, it can recognize and reproduce the internal structure of data from observations. So, this spatial dependency can affect data prediction accuracy using ANN. Unlike conventional methods, the neural network does not require a specific model such as Variogram, or conditions such as stationarity and isotropy.

Some existing studies have used a specific data set to compare between Kriging and Neural networks to verify which method has better performance (Matias et al., 2004; Gumus and Sen, 2013). Hybrid models for ANNs, based on the simultaneous use of ANN and geostatistical methods, are also expanded to model nonlinear spatial processes (Kanevsky et al., 1996; Yeh et al., 2013; Chung et al., 2012; Seo et al., 2015; Lin and Chen, 2004). However, these studies have been based on a specific data set. This paper aims to investigate, through a simulation study, the ability of artificial neural networks in learning the dependency structure of spatial data by comparing the network accuracy in predicting dependent and independent (or weakly dependent) spatial data. A simulation study is an invaluable tool in statistical studies,

especially in assessing and comparing competing methods. In a simulation study, the data are generated by pseudo-random sampling from known probability distributions (Morris et al., 2019; Ripley, 1987).
We also assessed the performance of ANN on an experimental data set contained the water quality of 122 wells based on electrical conductivity (EC) at Mashhad plain (Iran). This data set was previously used to estimate the electrical conductivity of groundwater with several nonlinear modeling techniques using the longitude and latitude of sampling points (Khashei-Siuki and Sarbazi, 2015).
This paper is organized as follows: Section 2 briefly describes the spatial data and the spatial dependency structure. In Section 3, a brief overview of the artificial neural network is presented. Section 4 describes the data simulation steps and reports the simulation results. In section 5, the ANN prediction accuracy was compared with Kriging based on a real data set. Finally, some conclusions are drawn in the final section.

2 Dependency Structure of Spatial Data

A collection of random variables $\{Z(s); s \in D \subset \mathbb{R}^d\}$ is defined as a d-dimensional *random field* (RF), in which the constituent random variables are correlated (Hristopulos, 2020; Diggle and Ribeiro, 2007). The observations of these random variables, called *spatial data*, represent a realization of the random field, and this is the basic foundation on which the statistical inferences are built for a spatially distributed variable (Zimmerman and Zimmerman, 1991). These observations include information about spatial coordinates (usually longitude and latitude), along with the value of the underlying variable. Usually, these observations are auto-correlated, in contrast with the most conventional data in classical statistics.

The spatial statistics and its sub-branch geostatistics, attempt to establish a relationship between various values of the spatially distributed variable and their distances and directions. For simplicity, it is usually supposed that the correlation of spatial data is a function of observations distances. The dependence structure of spatial data, under stationary condition, is estimated and modeled using *Variogram* or *Covariogram*. They measure the strength of correlation as a function of data distance. Suppose that $Z(s_1)$ and $Z(s_2)$ are random variables in two locations s_1 and s_2, respectively. Variogram and

Covariogram represent respectively the degree of *dissimilarity* and *similarity* of the variability for these two variables. They quantify the assumption that things nearby tend to be more similar than things that are farther apart.

The Variogram has a considerable advantage over the Covariogram because its estimation does not require knowledge of the expectation of the random field (Hristopulos, 2020). In practice, the Variogram is unknown and should be estimated using spatial data $Z(s_1), \ldots, Z(s_n)$. According to Matheron (1963), the Variogram can be estimated by $2\hat{\gamma}_h = \frac{1}{|N_h|}\sum_{N_h}\left(z(s_i) - z(s_j)\right)^2$, in which N_h is the set of all observations pairs within a lag distance h, $\mid N_h \mid$ represents the number of distinct pairs in N_h and $z(s_i)$ is data value at spatial location s_i. After estimating the Variogram, a valid parametric model should be fitted to it, such as Exponential $\gamma(\boldsymbol{h}) = c_0 + c_1\left(1 - \exp(\frac{-3\|\boldsymbol{h}\|}{r})\right)$ or Spherical model:

$$\gamma(\boldsymbol{h}) = \begin{cases} c_0 + c_1\left(\frac{3}{2}\frac{\|\boldsymbol{h}\|}{r} - \frac{1}{2}\left(\frac{\|\boldsymbol{h}\|}{r}\right)^3\right) & 0 < \|\boldsymbol{h}\| \leq r \\ c_0 + c_1 & \|\boldsymbol{h}\| > r \end{cases}$$

where $\boldsymbol{h} \in \mathbb{R}^d$ is the distance vector between two spatial locations. The parameters nugget effect (c_0), partial sill (c_1) and range (r), should be estimated using a method such as maximum likelihood, ordinary least squares, generalized least squares, or weighted least squares.

In the analysis of spatial data, the random field and their observations should satisfy conditions including stationarity, isotropy, and sometimes multivariable normality. These conditions simplify the estimation and modeling of Variogram and Covariogram. However, in practice, it is unlikely to satisfy these conditions. Also, checking these conditions require some expert knowledge of spatial statistics. Nonetheless, ANNs do not require that the spatial dependency has a specific structure and model. Also, ANNs can generalize nonlinear spatial patterns for the prediction at non-sampled locations.

3 Artificial Neural Network

Artificial neural network (ANN) is inspired by the biological nervous system to process information, initially proposed by McCulloch and Pitts (1943). ANN is typically composed of highly interconnected processing elements

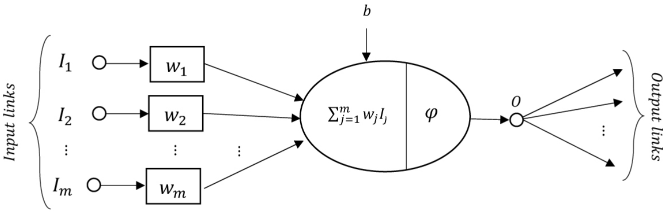

Figure 1. schematic illustration of an artificial neuron

called neurons (processing units) working together to solve a problem (Fig. 1).

Mathematically, the output O of a neuron is computed as:

$$O = \varphi\left(\sum_{j=1}^{m} w_j I_j + b\right) \tag{1}$$

where m is the total incoming links, I's are the inputs, w's are the weights, b is the bias, and $\varphi(.)$ is called the *activation* (or *transfer*) function. Weights are network parameters, and determine the effect of each input on the output within each neuron. Activation function, as an essential component of neural networks, gets a linear combination of inputs and limits its amplitude into a specified range. There are so many variations of this function which should be chosen by the user according to the problem. The most commonly used activation functions are the sigmoid $\varphi(x) = \dfrac{1}{(1+\exp(-x))}$, the tangent-hyperbolic $\varphi(x) = \tanh x$ and the linear function $\varphi(x) = x$.

An ANN usually could consist of several layers of these artificial neurons arranged in parallel, where the neurons in a layer have links from the neurons at the previous layer. There are two different ANN structures, which are Feed-Forward Neural Networks (FFNN) and Recurrent Neural Networks (RNN). In FFNN, the information always moves forward in the network (no loops), from the input layer, through the hidden layers, to the output layer. But in an RNN, the links between the neurons may form cycles (the information cycles through a loop). Multi-Layer perceptron (MLP) (Rosenblatt, 1958; Werbos, 1974) and Radial Basis Function (RBF) (Broomhead and Lowe, 1988; Poggio and Girosi, 1989) are the most commonly used FFNNs.

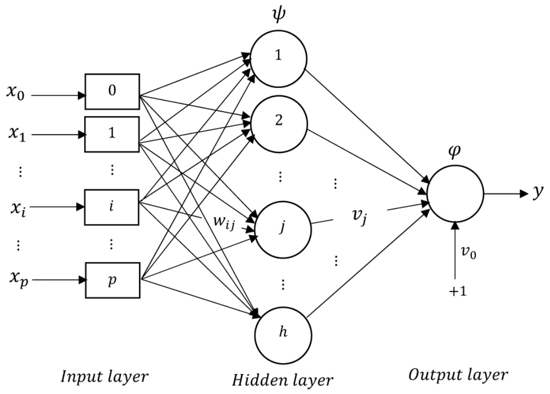

Figure 2. schematic illustration of MLP network with p inputs and one output

The process during which network parameters (weights) are estimated are referred to as *learning* or training process. The neural network utilizes an algorithm called *learning algorithm* to estimate the weights and biases by minimizing the error between the predicted and the actual response. In training process, experimental data (training data) is introduced to the neural network in order to establish a relationship between the input and output.

Although the other ANN models are capable of performing similar tasks, this paper is however limited only to MLP. This network consists of many processing units, organized in an input layer, an output layer, and several hidden layers. MLP is capable of solving different complicated problems using a widely used learning algorithm known as the *error back-propagation* (BP) algorithm (Rumelhart et al., 1986). Figure 2 shows an MLP network with p inputs, one output, and one hidden layer of h neurons. Mathematically, the MLP output is computed as $y = \varphi\left(\sum_{j=1}^{h} v_j \psi\left(\sum_{i=1}^{p} w_{ij} x_i + w_{0j}\right) + v_0\right)$, where $\boldsymbol{x} = \langle x_1, x_2, \ldots, x_p \rangle$ and y represent inputs and output respectively, and w and v present the network's weights. ψ and φ are activation functions for neurons of hidden and output layers, respectively. v_0 and w_0 are the bias terms where $x_0 = +1$.

The MLP is trained by providing a set of training data (X, Y) of n input-output pairs, i.e., $X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ and $Y = (y_1, \ldots, y_n)$, where each p-dimensional input vector $\boldsymbol{x}_i = \langle x_{i1} \ldots, x_{ip} \rangle$, $i = 1, \ldots, n$ is paired with corresponding desired output $y_i,\ i = 1, \dots, n$. The network produces an output $\hat{Y} = (\hat{y}_1, \dots, \hat{y}_n)$, which is then compared with the desired output $Y = (y_1, \dots, y_n)$ to minimize an error/cost function. This minimization task, which is done in an iterative manner, is referred to as *supervised learning*, in which for each input, the desired output (target value) is given to the network, and the network is trained to determine the input-output relationship.

The n-dimensional weight vector $\boldsymbol{w}$ of an MLP is optimized using delta rule (Widrow and Hoff, 1960) as follow:

$$\boldsymbol{w}^{t+1} = \boldsymbol{w}^t + \boldsymbol{\Delta w}^t,$$

where $\boldsymbol{\Delta w}^t$ is the weight update/change at t-th iteration and is calculated as

$$\boldsymbol{\Delta w}^t = \eta^t e_i^t \boldsymbol{x}_i^t,$$

where e_i^t is the error at t-th learning iteration corresponding to i-th training input $\boldsymbol{x}_i^t$ presented to MLP, and may be calculated as $e_i^t = (y_i - \hat{y}_i^t)$, where $\hat{y}_i^t$ is the network output at t-th iteration for i-th training output. The parameter η^t stands for learning rate, which controls the magnitude of weight change at t-th iteration and should be bounded in a small range (a small positive value, often in the range between 0.0 and 1.0) to ensure the stability of the algorithm. For a comprehensive overview of ANN techniques, computational problems, and its statistical interpretations, refer to (Bishop, 1995, 2006; Ripley, 2007; Ojha et al., 2017).

4 Simulation Study

As previously mentioned, ANN is a data-driven approach and does not directly estimate or model the form of spatial data dependency. To verify the network's ability in utilizing spatial dependency of data, its prediction accuracy was evaluated using simulated spatial data sets, with a total of 72 different scenarios (different auto-correlation structures and different sample sizes), and with 200 iterations for each scenario. In other words, the network prediction accuracy was compared for several independent (or weakly dependent) and dependent simulated spatial data sets. The network structure used in this study was MLP, and the computations were performed by the *nnet* package in **R** software.

4.1 Simulation of spatial data

To diversify the simulated data as well as the results, different scenarios were considered. Spatial data sets were simulated from two different spatial models: Gaussian random field (GRF) with a specific Variogram model, and first-order spatial autoregressive (SAR) model.

For the dependency structure of GRF model, an Exponential Variogram model i.e., $\gamma(h) = c_0 + c_1 (1 - \exp(-3h/r))$, was used with fixed Nugget effect ($c_0 = 0$), fixed partial sill value ($c_1 = 4$) and three values for the effective range parameter ($r = 0.01$, $r = 5$ and $r = 10$). Simulation of these spatial data was performed using the *grf* function of **R**'s *geoR* package. The reason for using three different values for the effective range parameter was to examine the effect of the spatial dependency of data on reducing the prediction error.

For the second model (SAR), the data were generated from a first-order spatial autoregressive model, *i.e.,* $Z = (I - \rho W)^{-1}\epsilon$, where $\epsilon \sim N_n(\mathbf{0}, \sigma^2 I)$, in a regular grid space (Mojiri et al., 2018). In this model, W is a weight matrix that determines the spatial dependency between spatial observations. Also, ρ is the spatial autocorrelation parameter, where three values ($\rho = 0$, $\rho = 0.4$ and 0.9) were considered in the simulation of spatial data.

For each of the GRF and SAR models, two cases were considered.

1) Spatial data set with constant mean (without trend). To simulate these data, the mean value was fixed at 100.

2) Spatial data set with non-constant mean (with trend). The trend was considered based on the following equation:

   $$\mu(x,y) = 100+0.04x-0.03y-0.07x^2+0.06y^2+0.06xy+2\sin(x)-3\cos(y)$$

   where coordinates x and y indicate the geographic location of the observations, generated on a regular grid of $[0,10] \times [0,10]$. The reason for using such an almost complicated trend was to show that MLP can learn and model the nonlinear and complex relationships between inputs and outputs.

To determine the effect of sample size on prediction accuracy, six different sample sizes of 64, 144, 196, 289, 400 and 625 were considered for each of the above-mentioned configurations. data included spatial coordinates x and y and the value $z(s_i)$, where $s_i = (x_i, y_i)$. The different 72 simulation scenarios can be seen in table 1 and 2.

4.2 the results of the simulation study

In this section, a summary of network performance results in predicting spatial data is separately reported for two previously mentioned models. The MLP network structure for all scenarios contained two inputs (x and y coordinates), an output (z), and one hidden layer. The optimal network structure (i.e., the number of hidden neurons and weight decay) was selected automatically from a range of values (0.001, 0.01, and 0.1 for weight decay and 3 to 30 for the number of hidden neurons), determined by a cross-validation algorithm written in **R** software. For simulated data from GRF, network performance was evaluated using the root mean square error criterion $RMSE = \sqrt{\frac{1}{n}\sum\left(z(s_i) - \hat{z}(s_i)\right)^2}$, which is a frequently used measure of the differences between predicted and observed values. However, for simulated data based on SAR, as the value of the dependency parameter ρ increases, the variance of the data also increases. Therefore, to eliminate the effect of variance and compare the results, the normalized root mean square error i.e., $NRMSE = RMSE/s_z$ was used instead of the RMSE criterion. For both criteria, the leave-one-out cross-validation technique (LOOCV) was used to estimate $\hat{z}$ values. In this technique, an observation is removed, and the network is trained using the remaining data. This trained network is then used to predict the value of the deleted observation. This process is repeated for each of the n observations, and finally, the RMSE or NRMSE is computed.

A: Results for GRF

The RMSE values for the GRF model is shown in figure 3 for a sample of size $n = 144$. This figure represents the RMSE values obtained from 200 simulation repetitions. The solid (red) points are related to the independent spatial data, and the triangular (blue) and hollow circular (black) points are respectively related to data with $r = 5$ (low dependency) and $r = 10$ (high dependency). According to the figure, in all repetitions, the network prediction error obtained for the dependent spatial data is strictly less than that of the independent spatial data. Also, the amount of error fluctuations is lower

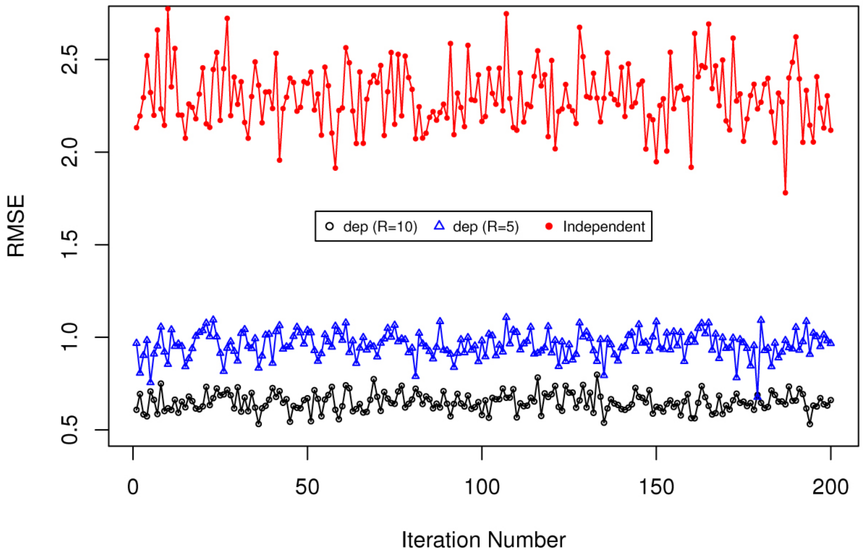

Figure 3. RMSE values for prediction of spatial data in GRF model

for dependent spatial data. The range of error variations is approximately 2, 1.11 and, 0.73, respectively for $r = 0.01$, $r = 5$ and $r = 10$.

Table 1 shows the general results of network performance for simulated data from GRF with the accuracy of three decimal places. The values in the table are the average RMSE values in 200 repetitions of the simulation. The mean values for the RMSE with their standard deviations are also presented in figure 4.

According to figure 4, the mean prediction error is reduced by increasing

Table 1. Average RMSE computed from 200 simulated data sets (GRF model)

Sample sizes	Constant mean			Non-constant mean		
	r=0.01	r=5	r=10	r=0.01	r=5	r=10
64	2.716	1.246	0.815	2.661	1.413	1.060
144	2.297	0.960	0.651	2.257	1.076	0.794
196	2.213	0.915	0.620	2.188	1.025	0.747
289	2.122	0.859	0.585	2.133	0.979	0.721
400	2.092	0.831	0.564	2.108	0.959	0.706
625	2.046	0.808	0.548	2.091	0.942	0.695

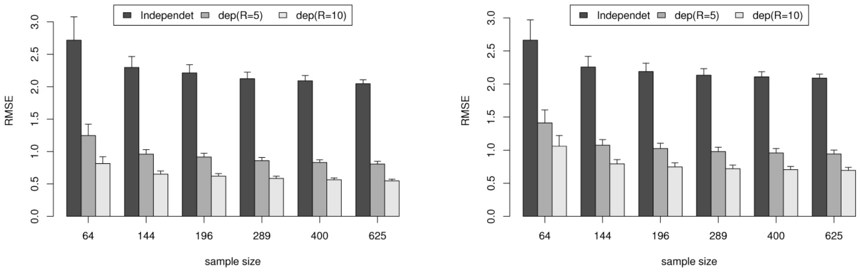

Figure 4. RMSE values for prediction of spatial data in GRF model (left: constant mean, right:Non-constant mean)

the data dependency (due to the increase in the range parameter). Also, increasing sample size led to a reduction of prediction error and similarly reduction of error variation. The reduction of network prediction error for dependent spatial data with *constant mean* indicated that neural network is able to make use of spatial dependency in prediction.

B: results for SAR model

Table 2 shows the mean values of NRMSE across 200 simulations for simulated data from the SAR model. As the results show, for the SAR model, the network prediction error for dependent spatial data is less than that of the independent case. However, error reduction for dependent data with the dependency parameter $\rho = 0.4$ is very small in constant mean data. Also, for non-constant mean data, the prediction error increased by increasing the sample size of dependent data. Therefore, unlike the GRF model, for data simulated from the SAR model, it can not be stated that network accuracy increases in spatial prediction by increasing the spatial dependency of the data (due to increasing in the value of ρ). However, in the SAR model, the prediction error of dependent spatial data is lower than that of independent spatial data.

5 Experimental Example

The simulated data in the previous section were well behaved with Gaussian residuals and a simple spatial dependency (simple Variogram). Real data are rarely so well behaved. Therefore in this section, an experimental data

Table 2. Average NRMSE computed for 200 simulated data sets (SAR model)

Sample sizes	Constant mean			Non-constant mean		
	$\rho = 0$	$\rho = 0.4$	$\rho = 0.9$	$\rho = 0$	$\rho = 0.4$	$\rho = 0.9$
64	1.196	1.189	0.846	0.407	0.324	0.316
144	1.083	1.052	0.712	0.375	0.294	0.292
196	1.058	1.022	0.692	0.370	0.290	0.289
289	1.038	1.007	0.697	0.365	0.292	0.294
400	1.028	1.003	0.719	0.364	0.296	0.308
625	1.017	0.998	0.740	0.364	0.297	0.324

Table 3. Schema of real data set

	longitude	Latitude	EC (μmos/cm)
i	(x_i)	(y_i)	$(Z(s_i))$
1	718151	4055410	663.0
2	705250	4041050	448.0
⋮	⋮	⋮	
122	695706	4020725	298.0

set is used to verify the prediction accuracy of ANN. This data set contains the Electrical Conductivity (EC) in the water of 122 wells at Mashhad plain. EC is an important indicator for water quality assessment. Mashhad plain, located in Northeast Khorasan Razavi Province, is extended between the longitudes of 58° 20 to 60° 8 and latitudes 35° 40 to 36° 3 of Iran. This data set, collected by the Iranian Ministry of Energy (IMOE), was measured from 21 April to 21 May 2009 (Khashei-Siuki and Sarbazi, 2015). The existence of the spatial autocorrelation in data was verified by the global Moran's index, producing a *p-value* of zero, indicating that the data are spatially dependent. So, it is expected that the ANN can use this dependency to have more accurate EC prediction.

Table 3 provides a preview of the data set, including the spatial location $s_i = (x_i, y_i)$ in UTM unit, along with the EC variable. The aim of this section is to predict EC for all locations using the MLP network. EC values and their locations are also shown in Fig. 5, where the circle size (diameter) is proportional to EC values.

Five-fold cross-validation (CV) was used to evaluate the prediction ac-

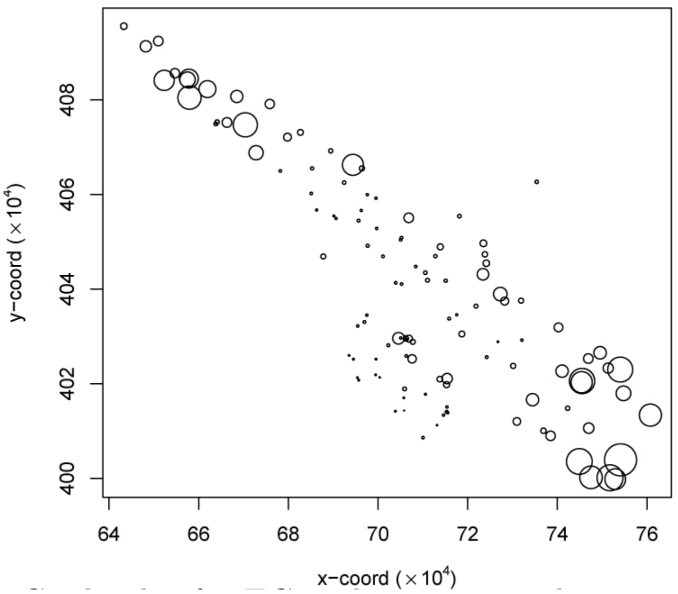

Figure 5. Circle plot for EC with centers at the spatial locations

curacy of the MLP network. The sampled data were randomly divided into five folds (partition) with approximately equal number of points. For each of five folds in the data set, MLP network was built and trained on the other four folds of the data set. Then, the trained MLP was verified on the remaining fold. This form of cross-validation technique was used (instead of leave-one-out), to investigate the capability of MLP in predicting over longer spatial ranges. The summary statistics of the EC variable at each fold are shown in table 4. We used *Neural Designer* software, version 4.2.0 for training the neural networks. For the MLP network structure, we considered an input layer, an output layer, and a hidden layer, where the input layer had two nodes according to longitude and latitude (geographical locations). The Minkowski error was used as the loss function that is more insensitive to outliers than the standard mean squared error. For the optimization algorithm, the Quasi-Newton method was selected with a maximum of 1000 iterations. The optimal topology of the network (optimum number of hidden nodes) was obtained automatically by the *incremental order* algorithm of Neural Designer software.

In addition, the prediction results of the Ordinary Kriging (OK), as a

Table 4. Performance indexes and descriptive statistics of predicted values

Folds	Method	performance indexes		summary statistics					
		RMSE	MAE	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
Fold1	ANN	1116.57	736.95	424.23	621.55	751.10	1156.8	1608.5	3952.2
	OK	1060.23	666.41	417.9	561.14	962.66	1290.8	1401.3	4831.8
			observed:	264	488.75	1193.5	1629.6	2245.0	4630.0
Fold2	ANN	851.12	477.54	249.4	426.36	652.08	1169.6	2144.9	2859.0
	OK	905.02	579.96	393.25	587.02	1143.38	1386.7	2028.1	3656.9
			observed:	282	423	686	1338.1	1844.7	4730.0
Fold3	ANN	783.79	578.85	272.67	490.9	977.170	1480.6	2190.4	4815.2
	OK	742.51	539.2	301.8	581.76	1104.45	1577.0	2627.3	4469.5
			observed:	169	511	898	1676.7	2315.0	5890.0
Fold4	ANN	697.87	503.78	265.45	441.11	719.32	1108.1	1661.3	3507.2
	OK	799.62	550.71	258.56	479.67	1065.45	1316.1	1673.7	3763.1
			observed:	284	502.25	844.9	1076.5	1518.3	3110.0
Fold5	ANN	763.35	513.26	253.84	531.6	797.01	1136.1	1471.9	3949.5
	OK	514.77	318.66	308.27	499.31	926.54	1309.9	1416.3	4623.7
			observed:	298	434	699	1220.8	1309.0	4650.0
Total	ANN	855.74	562.48	249.4	503.5	741.40	1210.9	1810.7	4815.2
	OK	826.59	532.77	258.6	530.9	1000.7	1376.7	1796.5	4831.8
			observed:	169.0	458.5	799.0	1389.8	1822.0	5890.0

common geostatistical method, are also provided in order to be able to compare the results. The OK implementation was done using the R's geoR package. The OK predictor estimates the value of the attribute using a linear combination of the observed values as $\hat{Z}(s_0) = \sum_{i=1}^{n} \lambda_i Z(s_i)$, where the weights λ_i are determined by minimizing the prediction variance, subject to $\sum_{i=1}^{n} \lambda_i = 1$ (Cressie, 1993).

The methods were evaluated using two performance indexes, root mean square error (RMSE) and mean absolute error (MAE) given in table 4, and also with box-and-whisker and scatter plots shown in figure 6. As table 4 shows, both indexes (RMSE and MAE) show that OK is generally more precise than ANN, with the values of 826.59 and 532.77 respectively for RMSE and MAE; But, the MLP method outperforms the OK method in two folds (fold 2 and fold 4), and for the other folds, MLP results are not significantly different from OK results. Also, by comparing the descriptive statistics of predicted and observed values in table 4 and data distribution in figure 6, it can be concluded that the MLP network was able to maintain the data distribution.

The scatter plot of predicted versus observed values are presented in figure 6 for OK and MLP methods. These plots describe how well the predicted values are close to the real values. The solid diagonal line indicates a

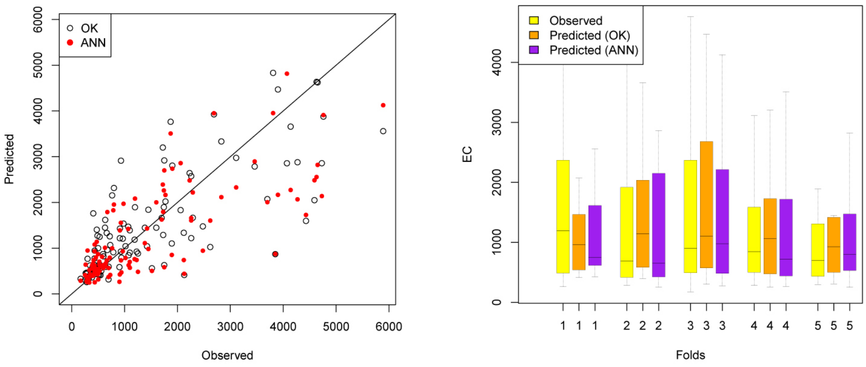

Figure 6. Observed versus predicted values of ANN and OK methods (left). The box plot of observed and predicted values (right)

perfect fit. Both methods produced good predictions for low values of EC, while for high values, the predicted values were somewhat overestimated or underestimated.

6 Conclusions and Discussion

In this paper, the precision of ANN predictions was investigated on three types of simulated spatial data: independent, weakly dependent, and highly dependent spatial data. The results showed that the network prediction error value for dependent spatial data is less than that of independent spatial data. This result was obtained regardless of whether the data had a trend or not. Given that the only difference between simulated independent and dependent spatial data was the presence or absence of spatial dependency between observations, we conclude that artificial neural network indirectly utilizes the spatial dependency in the prediction of spatial data, and data dependency has increased the accuracy of ANN prediction (due to reduction of prediction error).

The precision of ANN predictions was also investigated on experimental data and was compared with Kriging as a common geostatistical technique. According to the results, the prediction errors of MLP network were close to OK, with lower prediction error for 2 folds of 5 folds in a 5-fold cross-validation technique.

From the results of the simulation and experimental study, it concluded that the spatial prediction using artificial neural networks is a valuable technique that does not require the distinction between trend and covariance in the prediction process and does not have the limitations of Kriging methods such as the stationarity and isotropy assumptions. As a result, when the data does not have conditions like stationarity and isotropy, the neural network can be used as an alternative method for prediction.

It should be noted that in a particular problem, it is difficult to build an optimal network because of the wide range of network structures and learning algorithms. In this study, the optimal hyper-parameter of the MLP network was determined through an automatic algorithm, considering a range of values for each parameter. The inappropriate choice of the network topology can negatively affect the network performance on out-of-sample data. Also, expert knowledge is essential to select and prepare input data and to validate the results. The computing time required for network training is also too long, especially for large sample sizes. These problems affect the ANN's functionality and restrict its application for spatial prediction. Finally, a further simulation study is required to evaluate the prediction capability of ANN and to compare its accuracy with that of Kriging under different conditions.

References

Bishop, C. M. (1995). *Neural networks for pattern recognition.* Oxford university press.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning.* Springer-Verlag New York.

Broomhead, D. S. and Lowe, D. (1988). *Radial Basis Functions, Multi-variable Functional Interpolation and Adaptive Networks.* Technical report, Royal Signals and Radar Establishment Malvern (United Kingdom).

Carlin, B. P., Gelfand, A. E., and Banerjee, S. (2014). *Hierarchical Modeling and Analysis for Spatial Data.* Chapman and Hall/CRC.

Chung, C. H., Chiang, Y. M., and Chang, E. J. (2012). A Spatial Neural Fuzzy Network for Estimating Pan Evaporation at Ungauged Sites. *Hydrology and Earth System Sciences*, **16**, 255–266.

Cressie, N. A. C. (1993). *Statistics for Spatial Data. Wiley Series in Probability and Mathematical Statistics.* John Wiley & Sons, Inc.

Diggle, P. J. and Ribeiro, P. J. (2007). *Model-based Geostatistics.* Springer Series in Statistics.

Gelfand, A. E., Diggle, P., Guttorp, P., and Fuentes, M. (2010). *Handbook of Spatial Statistics.* CRC press.

Gumus, K. and Sen, A. (2013). Comparison of Spatial Interpolation Methods and Multi-layer Neural Networks for Different Point Distributions on a Digital Elevation Model. Geodetski Vestnik, pages 523–543.

Hristopulos, D. T. (2020). *Random Fields for Spatial Data Modeling.* Springer.

Kanevsky, M., Arutyunyan, R., Bolshov, L., Demyanov, V., and Maignan, M. (1996). Artificial Neural Networks and Spatial Estimation of Chernobyl Fallout. 7, 5–11.

Khashei-Siuki, A. and Sarbazi, M. (2015). Evaluation of ANFIS, ANN, and Geostatistical Models to Spatial Distribution of Groundwater Quality (Case Study: Mashhad Plain in Iran). *Arabian Journal of Geosciences*, **8**, 903–912.

Li, J. and Heap, A. D. (2011). A Review of Comparative Studies of Spatial Interpolation Methods in Environmental Sciences: Performance and Impact Factors. *Ecological Informatics*, **6**, 228–241.

Lin, G.-F. and Chen, L.-H. (2004). A Spatial Interpolation Method based on Radial Basis Function Networks Incorporating a Semivariogram Model. *Journal of Hydrology*, **288**, 288–298.

Matheron, G. (1963). Principles of geostatistics. *Economic Geology*, **58**, 1246–1266.

Matias, J. M., Vaamonde, A., Taboada, J., and Gonzalez-Manteiga, W. (2004). Comparison of Kriging and Neural Networks with Application to the Exploitation of a Slate Mine. *Mathematical Geology*, **36**, 463–486.

McCulloch, W. S. and Pitts, W. (1943). A Logical Calculus of the Ideas Immanent in Nervous Activity. *The Bulletin of Mathematical Biophysics*, **5**, 115–133.

Mojiri, A., Waghei, Y., Sani, H. N., and Borzadaran, G. M. (2018). Comparison of Predictions by Kriging and Spatial Autoregressive Models. *Communications in Statistics-Simulation and Computation*, **47**, 1785–1795.

Morris, T. P., White, I. R., and Crowther, M. J. (2019). Using Simulation Studies to Evaluate Statistical Methods. Statistics in Medicine.

Ojha, V. K., Abraham, A., and Snasel, V. (2017). Metaheuristic Design of Feedforward Neural Networks: A Review of Two Decades of Research. *Engineering Applications of Artificial Intelligence*, **60**, 97–116.

Poggio, T. and Girosi, F. (1989). A Theory of Networks for Approximation and Learning. Technical Report, Massachusetts Inst of Tech Cambridge Artificial Intelligence Lab.

Ripley, B. D. (1987). Stochastic Simulation, volume 5. Wiley Online Library.

Ripley, B. D. (2007). Pattern Recognition and Neural Networks. Cambridge University press.

Rosenblatt, F. (1958). The Perceptron: a Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, **65**, 386.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning Representations by Back-propagating Errors. *Nature*, **323**, 533–536.

Samarasinghe, S. (2016). *Neural Networks for Applied Sciences and Engineering: from Fundamentals to Complex Pattern Recognition*. Auerbach Publications.

Seo, Y., Kim, S., and Singh, V. P. (2015). Estimating Spatial Precipitation Using Regression Kriging and Artificial Neural Network Residual Kriging (RKNNRK) Hybrid Approach. *Water Resources Management*, **29**, 2189–2204.

Webster, R. and Oliver, M. A. (2007). *Geostatistics for Environmental Scientists*. Number 2. John Wiley & Sons.

Werbos, P. (1974). Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. Ph. D. dissertation, Harvard University.

Widrow, B. and Hoff, M. E. (1960). Adaptive Switching Circuits.

Yeh, I.-C., Huang, K.-C., and Kuo, Y.-H. (2013). Spatial Interpolation Using MLP–RBFN hybrid networks. *International Journal of Geographical Information Science*, **27**, 1884–1901.

Zimmerman, D., Pavlik, C., and Ruggles, A. (1999). An Experimental Comparaison of Ordinary and Universal Kriging and Distance Weighting. pdf. Mathematical Geology.

Zimmerman, D. L. and Zimmerman, M. B. (1991). A Comparison of Spatial Semivariogram Estimators and Corresponding Ordinary Kriging Predictors. *Technometrics*, **33**, 77–91.

Abbas Tavassoli
Department of Statistics,
University of Birjand,
Birjand, Iran.
email: *Tavassoli.a@birjand.ac.ir*

Yadollah Waghei
Department of Statistics,
University of Birjand,
Birjand, Iran.
email: *ywaghei@birjand.ac.ir*

Alireza Nazemi
Faculty of Mathematical Sciences,
Shahrood University of Technology,
Shahrood, Iran.
email: *nazemi20042003@gmail.com*