Artificial Neural Networks Based Approach for Identification of Unknown Pollution Sources in Aquifers

Maria Laura Foddis and Augusto Montisci(✉)

University of Cagliari, Cagliari, Italy
augusto.montisci@unica.it

Abstract. This work focuses on groundwater resources contaminations identification. The problem of identifying an unknown pollution source in polluted aquifers, based on known contaminant concentrations measurement in the studied areas, is part of the broader group of issues, called inverse problems. In this field, often pollution may result from contaminations whose origins are generated in different times and places where these contaminations have been actually found. To address such scenarios, it is necessary to develop specific techniques that allow to identify time and space features of unknown contaminant sources. The characterization of the contaminant source is of utmost importance for the planning of subsurface remediation in the polluted site. In this work, such identification is solved as an inverse problem in two stages. Firstly a Multi Layer Perceptron neural network is trained on a set of numerical simulations, and then the case under study is reconstructed by inverting the neural model.

Keywords: Artificial neural networks inversion · Inverse problems · Groundwater pollution source identification, groundwater modelling

1 Introduction

Only a small percentage of water present on earth is useful for human use and 98% of this water is represented by water reserves contained in aquifers. These reserves are the most important water resources used for agriculture, drinking and industrial purposes. However, groundwater is exposed to man-made pollution. One of the major issues for groundwater specialists is the effective management of the groundwater quality because contamination of groundwater may prevent its use. Due to increased pollution phenomena, groundwater has become increasingly vulnerable and its sustainable management is nowadays extremely important to protect global health (Foddis 2011; Smith et al. 2016; WHO 2017).

When groundwater is polluted the restauration of the quality and the removal of pollutants are a very slow, hence, lengthy, and, sometimes, practically impossible task. This implies the need to develop effective monitoring and pollution forecasting methods, which can support the protection of key zones, especially in those areas where the geological characteristics of the soil allow relatively easy penetration of

© Springer Nature Switzerland AG 2020
O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12255, pp. 877–890, 2020.
https://doi.org/10.1007/978-3-030-58820-5_63

anthropogenic pollution into the groundwater. Consequently, a sensible management and monitoring aimed at protecting the groundwater quality and at safeguarding the groundwater resources from contaminations have a vital importance for life support systems.

In the field of groundwater resources contaminations, it should be underlined that in some cases, pollution may be the consequence of contaminations whose origins are generated in different times and places from where these contaminations have been detected. To tackle such situations, the development of techniques that allow one to identify the features and the behaviour in time and space of these unknown pollution sources is compulsory.

In addition, the determination of the initial conditions of pollution at the contaminant source level is of considerable interest in the framework of the implementation of the European Union Directive 2004/35/EC: this directive concerns environmental liability with regard to the prevention and compensation of environmental damages, based on the "polluter-payer" principle.

In general, the problem of determining the unknown model parameters is usually identified in hydrogeology as "inverse problem" (Carrera et al. 2005). Solving the inverse problem, in hydrogeology, is the main goal of modelling groundwater flow and contaminant transport. In order to reach the solution of the inverse problem, in this work we propose the use of a new methodology based on the application of the Artificial Neural Networks (ANN) (Carcangiu et al. 2016, 2019; Yaman et al. 2013).

Over the past decades, ANNs have become increasingly popular as a problem-solving tool and have been extensively used as a forecasting tool in many disciplines. Many recent studies have focused on the use of ANNs to examine their suitability to model environmental processes, such as, soil geomechanical characterization (Secci et al. 2015), the effect of climate parameters with respect to groundwater levels (Jeihouni et al. 2019), the rainfall-runoff prediction (Tanty and Desmukh 2015), the prediction of processes in hydrologic cycle (Nourani et al. 2014), the prediction of nitrate concentration in groundwater (Ostad-Ali-Askari et al. 2016; Foddis et al. 2015b, 2017, 2019; Mousavi and Amiri 2012; Sathish Kumar 2013), the prediction of zonal transmissivity (Ajmera 2008) and the simulation of contaminant transport in porous media (Nourani et al. 2018, 2017a, 2017b).

Few authors investigated the feasibility of solving the inverse problems linked with hydrogeological phenomena by using ANNs. Among these, in Zio (1997), an ANN is trained to identify the value of the dispersion coefficient in a simple analytic contaminant transport model; in Fanni et al. (2002), an ANN is used to locate a pollutant source; in Rajanayaka et al. (2002), a hybrid approach based on a combination of two types of ANN model is applied to estimate hydrogeological parameters; in Mahar and Datta (2000) an ANNs is used to identify the location and duration of groundwater pollution sources under transient flow and transport conditions; in Scintu (2004), ANNs are trained to predict the coordinates of the pollutant source and the time the pollution occurred; in Singh and Datta (2007, 2004a, 2004b) ANNs are used to identify unknown pollution sources feature taking into account also cases where the concentration observation data were partially missing; in Foddis et al. (2013) are used for locating the source of a contamination event in time and space; in Foddis et al. (2015a) are used to

determine the profile of the pollutant source on the basis of a set of measurements in monitoring wells.

The purpose of this work is to demonstrate the feasibility of using the ANNs to define the behaviour of un unknown pollution sources. To this end, a theoretical scenario has been considered that consists of an accidental spill of a pollutant that caused the contamination of a shallow aquifer. The proposed ANN-based inverse problem-solving method can be summarised in two main steps. In the first step, an ANN is trained to solve the direct problem. After the training, the ANN generalization capability is exploited to estimate the contaminant concentration in pumping wells corresponding to a new pollution source. In the second step, the trained ANN is inverted in order to solve the inverse problem. In the following paragraph the methodology is deeply described.

2 Method Description

2.1 Training of the Neural Network Model

As highlighted in the previous section, ANNs have been widely used for the characterization of geological systems. The main advantage of this approach is that it is possible to study the behavior of the system without having an analytical model of it. Many different neural networks techniques are proposed in literature, depending on the specific problem at hand.

In particular, the Multi Layer Perceptrons (MLPs) ANNs used in this work, can be trained to imitate the system under study, seen as a dynamic input-output system. To this end, a suitable number of input-output pairs have to be generated to form the training set, and then the ANN is trained to associate the corresponding patterns. In this study, the Neural Network Toolbox of Matlab has been used.

The MLP is structured in layers of neurons (Fig. 1), the first being the Input Layer, and the last one the Output Layer. These are the only mandatory layers of the MLP. The intermediate layers (Hidden Layers) are the seat of the nonlinearity of the network, and they represent the degrees of freedom. In general, MLPs could have whichever number of layers, but it has been demonstrated that an MLP with only one hidden layer is a universal approximator (Cybenko 1989). For this reason, in this paper, the MLPs are considered having only one hidden layer without further specification. MLPs have the twofold advantage of using transcendent functions and of determining the parameters by means of examples. This second property makes it possible to develop a model of the system without an analytical formalization but simply based on a suitable set of input/output pairs of example patterns.

The features of the developed ANN depend on the nature of the analyzed problems and there are no theoretical guidelines for determining the best way out. The model is specific to the system under consideration and it cannot be built a priori.

The training of the ANN consists in applying a learning rule that modifies the weights of the connections based on the difference between the calculated and the desired output of the network. The aim of the training is to make the ANN able to

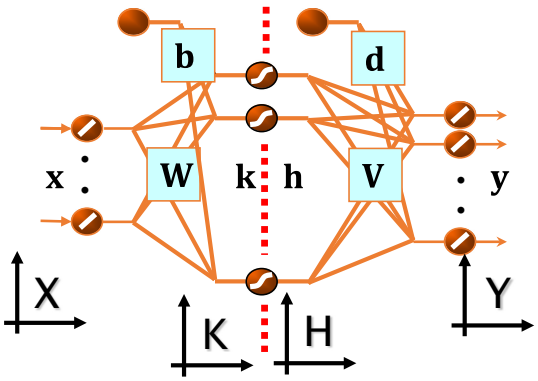


Fig. 1. Algebraic structure of the MLP neural network

generalize the acquired information, i.e. to give the correct output even for examples not included in the training set.

This aspect is crucial for the application described in this work, where the aim is to reconstruct the input by inverting the trained ANN.

2.2 Neural Network Model Inversion

The trained neural network has inside it all the required information to make a model of the input-output relationship. We can leverage on such information to solve the inverse problem, which consists in determining the contaminant source that correspond to a given evolution of the contaminant concentration in the pumping wells. To this end, let's consider the equations system that represents the algebraic input-output relation implemented by the neural network:

$$\begin{matrix} a) \\ b) \\ c) \end{matrix} \begin{cases} \mathbf{W} \cdot \boldsymbol{x} + \mathbf{b} = \boldsymbol{k} \\ \boldsymbol{h} = \sigma(\boldsymbol{k}) \\ \mathbf{V} \cdot \boldsymbol{h} + \mathbf{d} = \boldsymbol{y} \end{cases} \tag{1}$$

where capital letters represent matrices and lower case letters represent vectors, more in details $\boldsymbol{x}$ and $\boldsymbol{y}$ are the input and the output vectors respectively, $\boldsymbol{k}$ and $\boldsymbol{h}$ are auxiliary variables that respectively represent the input and the output of the hidden layer, $\mathbf{W}$ is the weights matrix of the first layer of connections and $\mathbf{b}$ is the corresponding bias vector, $\mathbf{V}$ is the weights matrix of the second layer of connections and $\mathbf{d}$ is the corresponding bias vector, and finally $\sigma(\cdot)$ is the monotonic nonlinear activation function of the hidden layer. From (1) it can be seen that the first and the third equations are linear, while the nonlinearity of the network is due the second equation, which is a set of decoupled invertible equations.

To handle the inversion problem, it is convenient to define the geometrical spaces where the single variables are described. Let's indicate with $\mathbf{X}$ the space where the input of the network is defined, $\mathbf{K}$ the input space of the hidden layer, $\mathbf{H}$ the output space of the hidden layer, and finally $\mathbf{Y}$ the output space of the whole neural network.

In Fig. 1, the defined parameters and variables are represented.

The objective of the inversion problem is to find the input vector $\mathbf{x}^*$ corresponding to a given output vector $\mathbf{y}^*$, namely the source of contaminant which corresponds to a given (measured) contaminant concentration time series at the pumping wells.

According to the precision and the sensitivity of the trained neural network, the target output $\mathbf{y}^*$ has to be assigned with a suitable margin. If the network is highly sensitive, a small deviation of the output could cause a large displacement of the inverted solution. On the other hand, if the model is not precise, a large margin has to be considered. Let's assume now that a proper margin has been determined, which combines the two conflicting requirements. Therefore, the inverse problem can be formulated as:

$$\text{find } \mathbf{x}^* \ni \mathbf{y}^* - \boldsymbol{\varepsilon} \leq \text{NN}(\mathbf{x}^*) \leq \mathbf{y}^* + \boldsymbol{\varepsilon} \tag{2}$$

where $\mathbf{y} = \text{NN}(\mathbf{x})$ is the output of the neural network calculated for the input vector $\mathbf{x}$, and $\boldsymbol{\varepsilon}$ is the tolerance of the output. The constraints in (2) are linear, and they can be globally expressed in the following simplified notation:

$$\mathbf{A} \cdot \mathbf{y} \leq \mathbf{a} \tag{3}$$

where $\mathbf{A}$ is the matrix of the coefficients and $\mathbf{a}$ is the constant term vector.

According to (1.c), the (3) can be reported in the space $\mathbf{H}$:

$$\mathbf{A} \cdot (\mathbf{V} \cdot \boldsymbol{h} + \mathbf{d}) \leq \mathbf{a} \tag{4}$$

At the same time, the vector $\boldsymbol{h}$ is the output of a saturating function (sigmoid), therefore, to be inverted its components must be within the interval of feasibility. Furthermore, a saturated value of the sigmoidal function makes the inversion undetermined, therefore the bounds of the $\boldsymbol{h}$ are assigned with a certain margin $\boldsymbol{\eta}$:

$$\mathbf{h}_{min} - \boldsymbol{\eta} \leq \boldsymbol{h} \leq \mathbf{h}_{max} - \boldsymbol{\eta} \tag{5}$$

Both (4) and (5) are linear constraints, therefore for sake of simplicity we will merge them into a unique linear inequalities system:

$$\mathbf{C} \cdot \boldsymbol{h} \leq \mathbf{c} \tag{6}$$

where $\mathbf{C}$ is the matrix of the coefficients and $\mathbf{c}$ is the constant term vector.

The inequality (6) represents the first check-point in inverting the target $\mathbf{y}^*$. In fact, in order to continue in backpropagating the target, the linear domain represented in (6) must be not empty. Well-known Linear Programming algorithms can be used to perform this check and, in case the domain is not empty, they return a feasible point of the domain. The possibility to find feasible points strictly depends on how the neural network has been trained. If the training set surrounds the target point, it is likely that the target can be obtained as the interpolation, although nonlinear, of the training set. In the present problem, the input-output relationship is nonlinear, therefore the training set could not have the proper characteristics at the first try. Nonetheless, resorting to the

inversion procedure, it is possible to generate new training examples that are more and more close to the target point, so that this is a feasible output of the network, and then the domain (6) is not empty.

Such feasible point can be translated from the space **H** to the space **K** by using the Eq. (1.b). Provided that the activation function of the hidden neurons is monotonic, the point $\boldsymbol{k}$, in the space **K**, corresponding to the feasible point is unique. The vector $\boldsymbol{k}$, together with the bias $\boldsymbol{b}$, plays the role of constant term in the linear equation system (1.a). Depending on the structure of the neural network, the (1.a) could be underdetermined (the hidden neurons are less than the input neurons), determined (equal number of neurons in the input and in the hidden layer), overdetermined (the input neurons are less than the hidden neurons). In most part of cases, the third case occurs, as the number of hidden neurons is directly connected to the degrees of freedom of the neural network. This occurred also in the present study, therefore it will be assumed as working hypothesis.

In the case of the coefficients matrix **W** in (1.a) is full-rank, the unique vector $\boldsymbol{x}$ which minimizes the minimum squared error can be assumed as solution of the system. This vector is the solution of the following linear equation system:

$$\mathbf{W}^{\mathbf{T}}\mathbf{W} \cdot \boldsymbol{x} = \mathbf{W}^{\mathbf{T}}(\boldsymbol{k} - \mathbf{b}) \tag{7}$$

The Eq. (7) is obtained by the (1.a) by multiplying per the transposal of the coefficients matrix, and it has equal number of unknowns and equations. By assuming that **W** is full-rank, the coefficients matrix $\left[\mathbf{W}^{\mathbf{T}}\mathbf{W}\right]$ is invertible, and then the solution can be determined as:

$$\boldsymbol{x} = \left[\mathbf{W}^{\mathbf{T}}\mathbf{W}\right]^{-1} \cdot \mathbf{W}^{\mathbf{T}}(\boldsymbol{k} - \mathbf{b}) \tag{8}$$

The vector $\boldsymbol{x}$ is the first proposal of the inversion problem solution $\mathbf{x}^*$. In general, the solution $\boldsymbol{x}$ of the (8) cannot match perfectly the vector $\boldsymbol{k}$, as the system is overdetermined. If the neural network is well-trained, we know that a vector $\boldsymbol{k}^*$ obtainable from $\mathbf{x}^*$ exists, but in case the hidden layer has more neurons of the output layer, there are infinite solutions of the Eq. (1.c), while if the neural network is well trained, the solution $\mathbf{x}^*$ of the inverse problem must be unique. In practice, one needs to find the intersection between the domain (6), which in the space **K** writes:

$$\mathbf{C} \cdot \sigma(\boldsymbol{k}) \leq \mathbf{c} \tag{9}$$

and the subspace of **K** generated by $\boldsymbol{x}$ by means of (1.a). Several procedures are available to find such intersection, which in terms of $\boldsymbol{x}$ writes:

$$\mathbf{C} \cdot \sigma(\mathbf{W} \cdot \boldsymbol{x} + \mathbf{b}) \leq \mathbf{c} \tag{10}$$

In this study, a simple first-order approach has been applied, which approximates the left-side hand of (10) with a linear function:

$$\mathbf{C} \cdot \sigma(\hat{\boldsymbol{x}}) + \mathbf{C} \cdot \mathbf{diag}[\sigma'(\hat{\boldsymbol{x}})] \cdot \mathbf{W} \cdot d\boldsymbol{x} \leq \mathbf{c} \tag{11}$$

where $\sigma'(\hat{\boldsymbol{x}})$ is the derivative of the activation function calculated in the current point, and $d\boldsymbol{x}$ is the increment of the variable. The (11) is used to seek iteratively one point of the domain (10), which will be a solution $\mathbf{x}^*$ of the inverse problem.

Remarks. In general, some trials are needed to obtain the required precision. This is mainly due to the fact that the training set at the beginning is not properly surrounding the target point, and then the trained neural network is not enough precise around that point. As a consequence, even if the output error of the neural network is within the margins, the uncertainty on the inverted solution $\mathbf{x}^*$ could be unacceptable. In this case, the pair $[\mathbf{x}^*, NN(\mathbf{x}^*)]$ can be added to the training set and the neural network trained again. By iterating this procedure, the training set becomes more and more focused on the target point, and then more precise in approximating the input-output relationship around that region.

A second remark concerns the sensitivity of the input with respect to the output. The forward robustness of the network, which is an advantage when the neural model is used for the solution of the direct problem, is a drawback when one is interested to solve the inverse problem, especially when the problem at hand is the parameter identification, like in the present application. In fact, the forward robustness implies that a wide range of inputs yields the same output, which means in the present study that the configuration of the contaminant source has large margins of uncertainty. A complete analysis of these aspects is beyond the scope of this work, and then it will be a topic for future works.

3 Case Study

The performance of the proposed methodology has been evaluated by defining the behavior in time and space of the unknown punctual pollution source of a generic phreatic aquifer.

3.1 Groundwater Flow and Contaminant Transport Numerical Model

The data set of the input/output pairs of example patterns was been constructed through a coherent number of hydrogeological scenarios, based on a 3D model of the domain developed by Aswed (2008). The input patterns were made of the pollution source features in terms of the injection rates in the four hydrogeological layers. The output patterns were contaminant concentration observation data at 45 pumping wells. Sources and pumping wells are related by a bi-univocal relationship, meaning that any specific profile of pumping wells corresponds to one specific contaminant source behavior.

The numerical model represents a contaminated zone enclosed within a 3D domain of 6 km width, 20 km length, and about 110 m depth. The aquifer domain was discretized by using a 3D triangular prismatic grid with 25,388 nodes and 45,460 elements. According to the estimated geometry of the cross-sections (the landfill site was divided into eight zones by soil type) the domain is discretized into ten layers. The

contaminant source is located in the first four layers having thicknesses, respectively, of 16 m, 4 m, 5 m, and 5 m from the top to the bottom. The volume of contaminated aquifer has been estimated in about 230–1,300 m^3 and the surface contaminant infiltration is about 7–37 m^2 assuming that the pollution depth is 35 m (Aswed 2008).

The 3D flux and contaminant transport numerical model used for constructing the patterns was calibrated using measured data of CCl4 concentration that were collected over 12 years (1992–2004) and simulations were performed for 1970 to 2024 (Vigouroux 1983; Aswed 2008).

To calculate the contaminant concentration in each pumping well we resort to the numerical simulation software TRACES (Transport or RadioActiver Elements in the Subsurface) developed by Hoteit and Ackerer (2003), that combines the mixed hybrid finite elements and discontinuous finite elements to solve the hydrodynamic state and mass transfer in the porous media.

The set of 292 input/output pairs of example patterns has been made derived from a random set of pollution sources behavior, and the corresponding contaminant concentration in pumping wells have been calculated by using TRACES.

The example patterns obtained with TRACES consist of 584 matrices of contaminant concentrations:

- 292 matrices corresponding to the features of the pollution sources. These has dimensions [11 520 × 4] where 11 520 represents the time (days) and 4 represents the layers in the source location.
- 292 matrices corresponding to contaminant concentration in the monitoring wells. These has dimensions of [4 000 × 45]; where 4 000 represents the values of contaminant concentration measured each 5 days (for a total time of simulation of 20.000 days) and 45 represents the monitoring wells in the domain. In this case it was taken only one value of contaminant concentration in monitoring wells each five days, for computational needs and this time step could not be increased, due to numerical criterion.

The 292 input matrices and 292 output matrices have been reorganized to form two matrices which describe the whole training set: one for the input and one for the output. Input and output matrices were too large to be processed through the ANN, so a data pre-processing has been performed in order to drastically reduce their dimension.

3.2 Training of the Neural Network Model

The system under study is the numerical model described in Sect. 3.1. The target was constructed by assuming a concentration of contaminant in the 4 layers under the origin for a period of 20 000 days (about 54 years). A homogeneous concentration is assumed in each layer, and also it is assumed that the concentration decays exponentially with the time. In this sense, the evolution of the source is completely described by 8 parameters, corresponding to 2 parameters of the exponential trend associated to each layer.

By means of the software TRACES, the propagation of the contaminant in the underground of the modeled area has been simulated for a period of time of 20 000 days with a time step of 5 days. The curve of the concentration has been then extracted in correspondence of the 45 pumping wells of the model (see Sect. 3.1).

In Fig. 2, the time diagram of the contaminant concentration in a well is reported. It looks like a Maxwell-Boltzmann curve, which can be assumed to approximate it. Such curve can be completely described with 3 parameters (occurrence of the max, maximum value and full width at half maximum), which can be used as features to describe the single curve. Nonetheless, the reconstruction of the time diagrams at the wells is not an objective of this study, therefore any set of features which univocally identify the curves of the dataset could be properly used. By comparing several curves obtained in the simulation it has been found that only 2 parameters, day of occurrence and averaged concentration, are sufficient to identify univocally the behavior at the wells. Taking into account that 45 pumping wells are foreseen in the case study, the pattern which represents the behavior of the wells in a single case consists of 90 components. This number should be strongly reduced to suitably train the neural network.

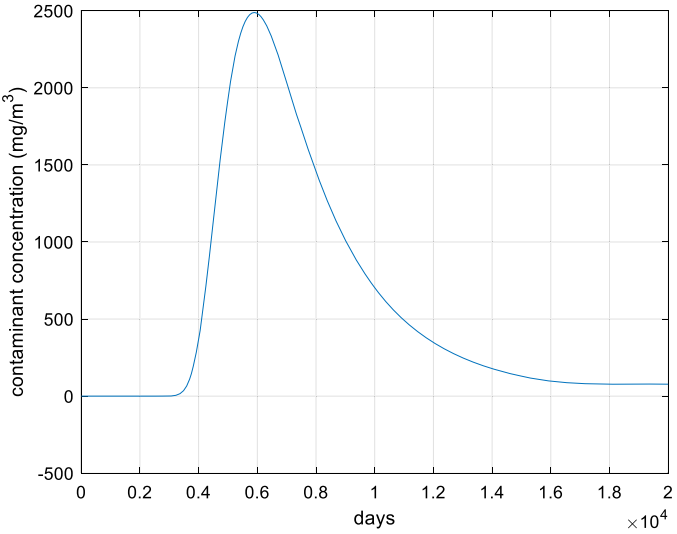


Fig. 2. Time diagram of contaminant measured at the pumping well

A training set of 292 examples has been created to train the neural network. To this end, a random set of sources has been created, and the corresponding evolutions have been calculated by means of TRACES. In the real cases, the creation of the training set is performed in a blind way, because only the measurements at the pumping wells are available. Therefore, it is possible that some outliers are included in the set, which could negatively affect the training. Such examples have to be detected and removed by the database. In order to reduce the volume of the data, the Principal Component Analysis has been applied to the 292 examples. As it is well known, the covariance matrix eigenvalues give a measure of the variance along each principal component, therefore it is possible to select a subset of components with a chosen variance. By dropping a fraction of variance of the distribution equal to 10^{-6}, the 90 components have been reduced to 6. On the basis of the distribution of these features, it is possible

to detect outliers, if there are any. In Fig. 3 the outliers detection method is shown. The first feature of the example No. 193 and the second feature of the example No. 51 are much different from the other ones, which implies that in normalizing the features the dynamic is lost, and the feature will be completely useless for the model. As a consequence, these examples have to be removed by the training set, and new examples have to be generated. A good strategy to define additional examples is to perform a convex combination of the examples whose output is near the target.

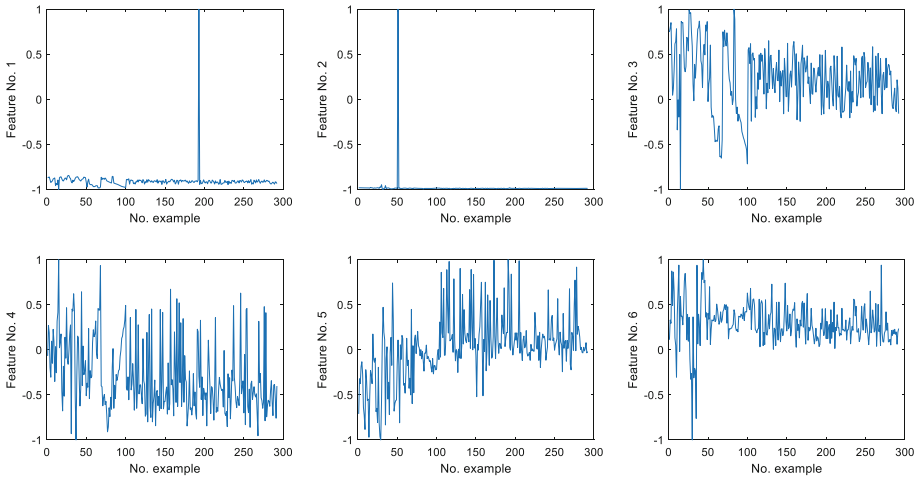


Fig. 3. Outliers detection by means of analysis of the features

The absence of outliers facilitates the training of the network. Several trainings, with different number of hidden neurons have been performed, in order to guarantee a good precision in reproducing the training set and in approximating the validation set. Finally, the structure assumed for the inversion has an 8-30-6 layout. In all the cases a number of epochs less than 100 have been sufficient to stabilize the value of the performance.

3.3 Neural Network Model Inversion

The inversion algorithm described in Sect. 2.2 has been applied to the trained neural network. An example belonging to the test set has been considered. The output domain (Eq. (2)) is defined as a neighborhood with margin 10^{-2} of the target to be inverted. This margin has been assumed on the basis of the error observed in the training set. In case no solution is found, this margin could be relaxed. The target vector and the margin completely define the feasibility domain of the output, according to Eq. (3). By means of the Eqs. (4) and (5), this domain is translated in the space **H**, then in the space **K** (Eq. (1.b)), and finally in the input space (Eq. (8)). By resorting to the Eq. (11), the solution has been iteratively modified in order to fulfil the constraints (2). The sought solution is the time diagram of the pollutant concentration in the four underground

layers. As the validation set has been created by running the FEM model, the exact solution is available for comparison. In the Fig. 4 the actual source is compared with that one obtained with the inversion procedure.

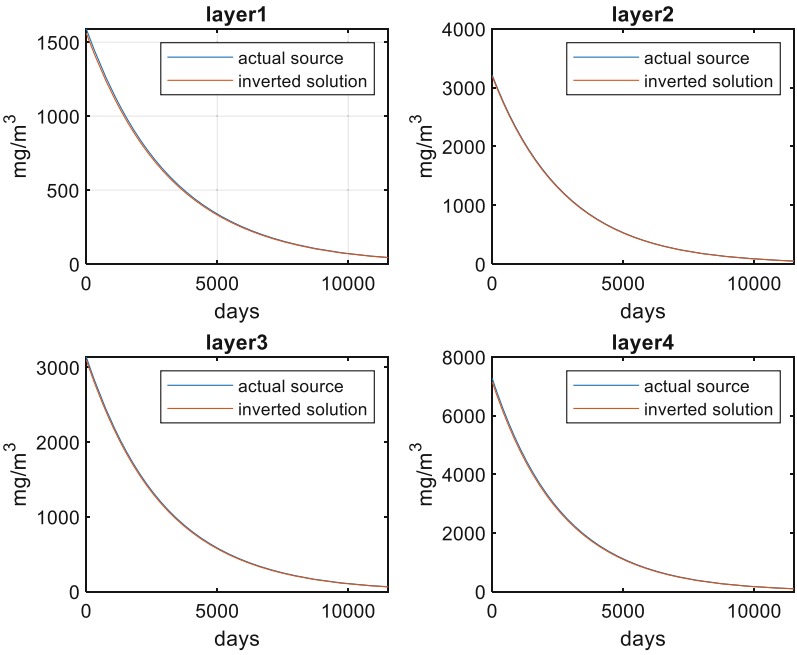


Fig. 4. Results of the inversion procedure

All the cases of training, validation and test sets assume by hypothesis an exponential trend in the source, and the neural model has been structured as a consequence. More specifically, the inputs of the neural network are the amplitude and the exponential coefficient of an exponential curve. For this reason, the exponential trend of the inverted solution is a result easy to obtain. To better evaluate the performance of the inversion it could be more meaningful to evaluate the percentage error of the inverted solution with respect to the exact one. In Table 1 this comparison is reported.

Table 1.

Solutions	Layer 1		Layer 2		Layer 3		Layer 4	
	Amp.	Coeff. $\times 10^{-3}$	Amp.	Coeff. $\times 10^{-3}$	Amp.	Coeff. $\times 10^{-3}$	Amp.	Coeff. $\times 10^{-3}$
Sought	1561	−3.10	3190	−3.59	3100	−3.38	7150	−3.76
Inverted	1590	−3.09	3213	−3.57	3139	−3.38	7270	−3.75
Err%	1.84	0.45	0.72	0.66	1.24	0.13	1.67	0.24

4 Conclusion

In this paper, a neural network-based method is proposed to solve the inverse problem of determining the source of contaminant of groundwaters from the measurements on a set of wells distributed in the area of interest. A neural network is first trained to solve the direct problem, namely determining the time diagram of the contaminant at the wells given the characteristics of the contaminant source. To this end, a suitable number of scenarios have been simulated by means of the software TRACES and the corresponding time evolution of contaminant concentration at the wells is determined. Successively, a scenario represented by time diagrams measured at the wells is presented to the previously trained neural network, and the corresponding input is determined. This input represents the solution of the inverse problem.

Acknowledgements. Authors would like to thank Prof. Philippe Ackerer from Laboratory of HYdrology and GEochemistry of Strasbourg (LHyGes - UMR 7517) of Strasbourg (France) for allowing the use of the TRACES software.

References

Ajmera, T.K., Rastogi, A.K.: Artificial neural network application on estimation of aquifer transmissivity. J. Spatial Hydrol. **8**(2), 15–31 (2008)

Aswed, T.: Modélisation de la pollution de la nappe d'alsace par solvants chlores. Ph.D. thesis, Université Louis Pasteur, Institut de Mécanique des Fluides et des Solides, UMR-CNRS 7507, Strasbourg, France (2008)

Carcangiu, S., Cardelli, E., Faba, A., Fanni, A., Montisci, A., Quondam, S.: Moving vector hysteron model identification based on neural network inversion. In: 2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a Better Tomorrow, RTSI 2016, art. no. 7740638 (2016)

Carcangiu, S., Fanni, A., Montisci, A.: Electric capacitance tomography for nondestructive testing of standing trees. Int. J. Numer. Model. **32**, e2252 (2019). https://doi.org/10.1002/jnm.2252

Carrera, J., Alcolea, A., Medina, A., Hidalgo, J., Slooten, J.: Inverse problem in hydrogeology. Hydrogeol. J. **13**, 206–222 (2005)

Cybenko, G.: Approximation by superposition of a sigmoid function. Math. Control Signals Syst. **2**, 303–314 (1989)

Directive 2004/35/CE European Parliament and of the Council of 21 April 2004 on environmental liability with regard to the prevention and remedying of environmental damage. Official Journal of the European Communities L 143/56, 30 April 2004

Fanni, A., Uras, G., Usai, M., Zedda, M.K.: Neural Network for monitoring. Groundwater. In: Fifth International Conference on Hidroinformatics, Cardiff, UK, 1–5 July 2002, pp. 687–692 (2002)

Foddis, M.L.: Application of artificial neural networks in hydrogeology: identification of unknown pollution sources in contaminated aquifers. Ph.D. thesis, University of Cagliari and University of Strasbourg (2011)

Foddis, M.L., Ackerer, P., Montisci, A., Uras, G.: Ann-based approach for the estimation aquifer pollutant source behaviour, water science and technology. Water Sci. Technol.: Water Supply **15**(6), 1285–1294 (2015a)

Foddis, M.L., Matzeu, A., Montisci, A., Uras, G.: Application of three different methods to evaluate the nitrate pollution of groundwater in the Arborea plain (Sardinia - Italy). Rendiconti Online Società Geologica Italiana **35**, 136–139 (2015b)

Foddis, M.L., Ackerer, P., Montisci, A., Uras, G.: Polluted aquifer inverse problem solution using artificial neural networks. AQUA Mundi **Am07054**, 015–021 (2013)

Foddis, M.L., Montisci, A., Trablesi, F., Uras, G.: An ANN-MLP based approach for the estimation of nitrate contamination. Water Sci. Technol.: Water Supply **19**(7), 1911–1917 (2019)

Foddis, M.L., Matzeu, A., Montisci, A., Uras, G.: The Arborea plain (Sardinia-Italy) nitrate pollution evaluation. Italian J. Eng. Geol. Environ. (Specialissue1), 67–76 (2017)

Hoteit, H., Acherer, P.: TRACES user's guide V 1.00. Institut mécanique des fluides et des solides de Strasbourg (2003)

Jeihouni, E., Eslamian, S., Mohammadi, M., Zareian, M.J.: Simulation of groundwater level fluctuations in response to main climate parameters using a wavelet–ANN hybrid technique for the Shabestar Plain, Iran. Environ. Earth Sci. **78**(10), 293 (2019)

Mahar, P.S., Datta, B.: Identification of pollution sources in transient groundwater systems. Water Resour. Manag. **14**, 209–227 (2000)

Mousavi, S.F., Amiri, M.J.: Modelling nitrate concentration of groundwater using adaptive neural-based fuzzy inference system. Soil Water Resour. **7**(2), 73–83 (2012)

Nourani, V., Mousavi, S., Sadikoglu, F.: Conjunction of artificial intelligence-meshless methods for contaminant transport modeling in porous media: an experimental case study. J. Hydroinformatics **20**(5), 1163–1179 (2018)

Nourani, V., Mousavi, S., Dabrowska, D., Sadikoglu, F.: Conjunction of radial basis function interpolator and artificial intelligence models for time-space modeling of contaminant transport in porous media. J. Hydrol. **548**, 569–587 (2017a)

Nourani, V., Mousavi, S., Sadikoglu, F., Singh, V.P.: Experimental and AI-based numerical modeling of contaminant transport in porous media. J. Contam. Hydrol. **205**, 78–95 (2017b)

Nourani, V., Hosseini, B.A., Adamowski, J., Kisi, O.: Applications of hybrid wavelet–artificial intelligence models in hydrology. J. Hydrol. **514**, 358–377 (2014)

Ostad-Ali-Askari, K., Shayannejad, M., Ghorbanizadeh-Kharazi, H.: Artificial neural network for modeling nitrate pollution of groundwater in marginal area of Zayandeh-rood River, Isfahan, Iran. KSCE J. Civil Eng. **21**(1), 134–140 (2016)

Rajanayaka, C., Samarasinghe, S., Kulasiri, D.: Solving the inverse problem in stochastic groundwater modelling with artificial neural networks. In: Rizzoli, A.E., Jakeman, A.J. (eds.) Integrated Assessment and Decision Support. International Environmental Modelling and Software Society, Manno, Switzerland, vol. 2 (2002)

Sathish Kumar, S., Mageshkumar, P., Santhanam, H., Stalin John, M.R., Amal Raj, S.: A new logic-based model to predict nitrates in groundwater using Artificial Neural Network (ANN). Pollution Res. **32**(3), 635–641 (2013)

Scintu, C.: Reti neurali artificiali: una applicazione nello studio di acquiferi contaminati. Ph.D. thesis, University of Cagliari, Italy (2004)

Secci, R., Laura Foddis, M., Mazzella, A., Montisci, A., Uras, G.: Artificial neural networks and kriging method for slope geomechanical characterization. In: Lollino, G., et al. (eds.) Engineering Geology for Society and Territory - Volume 2, pp. 1357–1361. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-09057-3_239

Singh, R.M., Datta, B.: Artificial neural network modeling for identification of unknown pollution sources in groundwater with partially missing concentration observation data. Water Resour. Manag. **21**(3), 557–572 (2007). https://doi.org/10.1007/s11269-006-9029-z

Singh, R.M., Datta, B.: Groundwater pollution source and simultaneous parameter estimation using pattern matching by artificial neural network. Environ. Forensics **5**(3), 143–153 (2004)

Singh, R.M., Datta, B., Jain, A.: Identification of unknown groundwater pollution sources using artificial neural networks. J. Water Resour. Plann. Manag. **130**(6), 506–514 (2004)

Smith, M., Cross, K., Paden, M., Laben, P.: Spring - managing groundwater sustainably. IUCN (2016). ISBN 978-2-8317-1789-0

Tanty, R., Desmukh, T.S.: Application of artificial neural network in hydrology-a review. Int. J. Eng. Tech. Res. **4**(6), 184–188 (2015)

Vigouroux, P., Vançon, J.P., Drogue, C.: Conception d'un model de propagation de pollution en nappe aquifer-Exemple d'application à la nappe du Rhin. J. Hydrol. **64**(1–4), 267–279 (1983)

World Health Organization (WHO): Protecting Groundwater for Health - Understanding the drinking-water catchment (2017)

Yaman, F., Yakhno, V., Potthast, R.: A survey on inverse problems for applied sciences. Math. Problems Eng. (2013). https://doi.org/10.1155/2013/976837

Zio, E.: Approaching the inverse problem of parameter estimation in groundwater models by means of artificial neural networks. Progress Nuclear Energy **31**(3), 303–315 (1997)