Neural networks for emulation variational method for data assimilation in nonlinear dynamics

Helaine Cristina Morais Furtado

Haroldo Fraga de Campos Velho

Elbert E. N. Macau

Laboratório Associado de Computação e Matemática Aplicada, São José dos Campos, Brasil
E-mail: helaine.furtado@lac.inpe.br, haroldo@lac.inpe.br, elbert@lac.inpe.br

Abstract. Description of a physical phenomenon through differential equations has errors involved, since the mathematical model is always an approximation of reality. For an operational prediction system, one strategy to improve the prediction is to add some information from the real dynamics into mathematical model. This additional information consists of observations on the phenomenon. However, the observational data insertion should be done carefully, for avoiding a worse performance of the prediction. Technical data assimilation are tools to combine data from physical-mathematics model with observational data to obtain a better forecast. The goal of this work is to present the performance of the Neural Network Multilayer Perceptrons trained to emulate a Variational method in context of data assimilation. Techniques for data assimilation are applied for the Lorenz systems; which presents a strong nonlinearity and chaotic nature.

1. Introduction

Data assimilation is nowadays prevailing process in the Numerical Weather Prediction (NWP) [1, 2], ocean circulation forecasting [3], air monitoring [4] and space weather models [5, 6]. Technical data assimilation can be defined as a set of methods that can be used to combine data from physical-mathematics model with observational data to obtain a better forecast.

From filtering point of view, data assimilation process can be represented by:

\[ w^a = w^f + Kp[y - H(w^f)] \]  (1)

where \( w^a \) is the value of the analysis; \( w^f \) is the forecasting (from the mathematical model, also known as background field); \( K \) is the weight matrix; \( y \) denotes the observation; \( H \) represents the observation system; the difference \( \{y - H(w^f)\} \) is the innovation; and \( p[.] \) is a discrepancy function. Another approach for solving the data assimilation is by computing the minimum solution for the cost function:

\[ J(w) = \langle (w - w^f), C(w - w^f) \rangle + \langle [y - H(w^f)], S[y - H(w^f)] \rangle \]  (2)
where $C$ and $S$ denote the inverse covariance matrices of the background (in general such matrix is represented by $B$ or $Q$ for the data assimilation or for the control theory communities, respectively) and of the measurement errors (denoted by $R$), and $\langle u, v \rangle$ expresses the internal product between vectors $u$ and $v$. For the Kalman filter (KF), the matrix $K$ (Kalman gain) at equation (1) is computed from a formula involving matrices $B$ and $R$. Variational scheme is linked to the optimization problem with the objective function given by expression (2).

In general, it is not easy the process of estimating the background matrix $B$ (also called the modeling error covariance matrix). There are many techniques to estimate this matrix: using some type of parameterization [7], employing ensemble KF [8], or a Fokker-Planck equation [9, 10]. For applying the KF, in general some linearization is necessary, and the random variables are assumed to have Gaussian distribution.

The process of data assimilation can be seen as a problem of determining the initial condition (i.c.) and defined as: “The science of having an appropriate combination of data from a mathematical model with observation data to determine the data analysis (or i.c.)” [11]. Therefore, a better agreement with the reality for the i.c. implies in a better forecasting. Figure 1 illustrates the technique of data assimilation. Note that whilst observations are inserted into the system, the dynamics of the estimation (red curve) is near to the truth (blue curve), since the data assimilation process is interrupted, there is a decoupling of the dynamics. Therefore, observations, when combined effectively with data from mathematical model, serve to bring the estimation of the initial condition to the desired value. In this work, synthetic observations are determined by integrating the Lorenz system adding some noise.

![Figure 1](image.png)

**Figure 1.** Time series of $x$ component Lorenz’s system. Blue curve: truth; green squares: observations; red curve: estimate and black curve: background. Adapted [12]

For considering a more general problem (nonlinear models and non-Gaussian distribution) the particle filter technique has been proposed [13]. However, this approach has a greater computational complexity than extended KF (EKF). The latter technique has a similar complexity to the 4-dimensional variational (4D-Var) scheme described in this work at Section 3.1.

Recently, we have introduced a new approach based on artificial neural network (ANN) for data assimilation [14, 15, 16, 17]. In this former articles, the ANN emulates the KF. For the present paper, the data assimilation by variational method is emulated by a supervised artificial neural network [12]. It is important to mention that after training ANN has a lower computational cost than extended and linear KF, variational method, and particle filter. The Lorenz system under chaotic regime will be used to illustrate the method.
The Lorenz system under chaotic regime is described in Section 2 will be employed for testing the assimilation schemes.

2. Testing model: Lorenz system

The dynamical Lorenz system is a simplified Saltzman’s model [18]. The celebrated Lorenz model has been employed as a standard test for examining the performance of data assimilation methods. The Lorenz nonlinear system is expressed as:

\[
\begin{align*}
\frac{dx}{d\tau} &= -\sigma (x - y) \\
\frac{dy}{d\tau} &= \rho x - y - xz \\
\frac{dz}{d\tau} &= \rho xy - \beta z
\end{align*}
\]

where \(\tau \equiv \pi^2 H^{-2}(1+a^2)\kappa t\) is the non-dimensional time (being \(H, a, \kappa\), and \(t\) layer height, thermal conductivity, wave number (diameter of the Rayleigh-Bénard cell), and time, respectively), \(\sigma \equiv \kappa^{-1} \nu\) is the Prandtl number (\(\nu\) is the kinematic viscosity); \(\beta \equiv 4(1 + a^2)^{-1}\). The parameter \(\rho = R/R_c \propto \Delta T\) is the Rayleigh number (\(T\) is the temperature), and \(R_c\) is the critical Rayleigh number. The variables \(x, y\) and \(z\) are the coefficients of time-dependent quantities from a partial differential equation of the atmosphere under heat source (convective motion, and the temperature gradients in the horizontal and vertical directions, respectively). The numerical solution this model results on the attractor shown by Figure 2.

![Lorenz attractor](image)

Figure 2. Lorenz attractor with initial conditions \((-1.5, 1.5, 20)\) and its projections on the axes \(xy, xz\) and \(yz\).

3. Data assimilation schemes

3.1. Variational method

This scheme focuses to compute the solution of an optimization problem. The objective function includes a term measuring the distance to the background at the beginning of the interval, and a summation over time of the objective function for each observation increment computed with respect to the model integrated to the of the observation [2]:

\[
J[x(t_0)] = \frac{1}{2} [x(t_0) - x^b(t_0)]^T B_0^{-1} [x(t_0) - x^b(t_0)] + \frac{1}{2} \sum_{i=0}^{N} [H(x_i) - y_i]^T R_i^{-1} [H(x_i) - y_i]
\]  

(6)
The control variable \( x \) is the initial state of the model. The time integration of the dynamical system from the initial condition up to the time \( t_n \) could be represented by 
\[
x(t_n) = M_0[x(t_0)]
\]
where the gradient of the cost function \( \frac{\partial J}{\partial x(t_0)} \) is a column vector. To obtain the minimum functional \( J \) by an iterative method, it is necessary to integrate the adjoint model \((M_i)'\), described in section 3.2.

The gradient of the first term of functional \( J_b \) of 6 with respect to \( x(t_0) \) is given by:
\[
\frac{\partial J_b}{\partial x(t_0)} = B_0^{-1}[x(t_0) - x^b(t_0)] .
\]  

(8)

The gradient of the second term \( J_o \) of functional 6 is more complicated due \( x_i = M_i[x(t_0)] \), where \( M \) represents the model prediction. If we introduce a disturbance to the initial state, then \( \delta x_i = L(t_0, t_i)\delta x_0 \), so that:
\[
\frac{\partial (H(x_i) - y_i^o)}{\partial x(t_0)} = \frac{\partial H}{\partial x_i} \frac{\partial x_i}{\partial x_0} = H_i L(t_0, t_i) = H_i \prod_{j=i-1}^0 L(t_j, t_{j+1}) .
\]  

(9)

The arrays \( H_i \) and \( L_i \) are linearized Jacobian \( \frac{\partial H}{\partial x_i} \) and \( \frac{\partial M}{\partial x_0} \). \( L \) is the matrix that propagates an initial disturbance to final time of integration (tangent linear model) [19].

Therefore, the gradient of the term of observation is given by:
\[
\frac{\partial J_o}{\partial x(t_0)} = \sum_{i=0}^N L(t_i, t_0)^T H_i^T R_i^{-1} [H(x_i) - y_i^o] .
\]  

(10)

Equation (10) shows that each iteration of the minimization of this functional requires the calculation of the gradient, i.e., to calculate the increment \( [H(x_i) - y_i^o] \) at the observation time \( t_i \) during the advanced integration, multiplying them by \( H_i^T R_i^{-1} \) and integrating these weighted increments back to the initial time using the adjoint model (transposed tangent linear model). Since the adjoint integration is common in several time intervals, the sum 10 can be rearranged more conveniently (see [2]).

3.2. Tangent linear model and adjoint model

Consider a nonlinear model. Since this model must be discretized in space using finite differences, the model can be written as a set of \( n \) coupled differential equations:
\[
\frac{dx}{dt} = F(x), \quad x = [x_1 \ldots x_n] \quad \text{e} \quad F = [F_1 \ldots F_n]
\]  

(11)
This is the model in differential form. An atmospheric model consists of one system of difference equations which, for example, using the Crank-Nicholson method can be described of the form:

\[ x^{n+1} = x^n + \Delta t F \left( \frac{x^n + x^{n+1}}{2} \right) \]  (12)

A numerical solution of 11 starting from an initial time \( t_0 \) can be obtained by integrating the model numerically using 12 between initial time and a final time. This provides us a nonlinear model solution that depends only on the initial conditions:

\[ x(t) = M[x(t_0)] \]  (13)

where \( M \) is the time integration of the numerical scheme from the initial condition up to time \( t \). A small perturbation \( y(t) \) can be added to the basic model integration \( x(t) \):

\[
M[x(t_0) + y(t_0)] = M[x(t_0)] + \frac{\partial M}{\partial x} y(t_0) + O[y(t_0)^2] \\
= x(t) + y(t) + O[y(t_0)^2] \\
\]  (14)

At any given time, the linear perturbation of \( y(t) \) will be given by:

\[
\frac{dy}{dt} = Jy \\
\]

where \( J = \frac{\partial F}{\partial x} \) is the Jacobian of \( F \).

This system of linear ordinary differential equations is the tangent linear model in differential form. Its solution between \( t_0 \) and \( t \) can be obtained by integration of (15) in time using the same time difference scheme used in the nonlinear model (11):

\[
y(t) = L(t_0, t)y(t_0) \\
\]  (16)

Here \( L(t_0, t) = \frac{\partial M}{\partial x} \) is an \( m \times m \) matrix known as the matrix of the tangent linear model, it propagates an initial perturbation at time \( t_0 \) into the final perturbation at time \( t \). Lorenz (1965) [19] introduced the concept of tangent linear model of an atmospheric model, obtained from 14, neglecting quadratic or higher order terms in the perturbation \( y \):

\[
M[x(t_0)] + L(t_0, t)y(t_0) = x(t) + y(t) \approx M[x(t_0) + y(t_0)] . \\
\]  (17)

Adding a small perturbation of size \( \varepsilon \) along the vectors \( y_i(t_0) = \varepsilon e_i \) and applying (17) to each of these perturbations and subtracting (13) obtain the matrix that defined the tangent linear model:

\[
L(t_0, t)[\varepsilon e_1, \ldots, e_n] = \varepsilon L(t_0, t) = [y_1(t), \ldots, y_n(t)] . \\
\]  (18)

The Euclidean norm of a vector is the inner product of the vector with itself:

\[
\|y\|^2 = y^T y = \langle y, y \rangle . \\
\]  (19)

The Euclidean norm of \( y(t) \) is related to the initial perturbation by:

\[
\|y(t)\|^2 = \langle Ly(t_0), y(t_0) \rangle = \langle Ly(t_0), Ly(t_0) \rangle = \langle L^T Ly(t_0), y(t_0) \rangle \\
\]  (20)

The adjoint of an operator \( K \) is defined by the property \( \langle x, Ky \rangle = \langle K^T x, y \rangle \). In this case of a model with real variables, the adjoint tangent linear model \( L(t_0, t) \) is simply the transpose of the tangent linear model.
Now assume that we separate the interval \((t_0, t)\) into two successive time intervals. For example, if \(t_0 < t_1 < t\),

\[
L(t_0, t) = L(t_1, t)L(t_0, t_1).
\]  
(21)

Since the adjoint of tangent linear model is the transpose of it, the property of the transpose of a product is also valid:

\[
L^T(t_0, t) = L^T(t_0, t_1)L^T(t_1, t).
\]  
(22)

Equation (21) shows that the tangent linear model can be computed with a product of the tangent linear model matrices corresponding to short integrations. From equation (22), the adjoint model can be separated into single time steps, but they are executed backwards in time, starting from the last time step at \(t\), and ending with the first time step at \(t_0\). For more explanations (see [2]).

3.3. Artificial neural networks

Artificial Neural Networks (ANN) have become important tools for information processing [20]. Much research has been conducted in pursuing new neural network models and adapting the existing ones to solve real life problems, such as those in engineering [20]. ANN are made of arrangements of processing elements called neurons. The artificial neuron model basically consists of a linear combiner followed by an activation function, Figure 1 (left side), given by:

\[
y_k = \varphi \left( \sum_{j=1}^{n} w_{kj} x_j + b_k \right)
\]  
(23)

where \(w_{kj}\) are the connection weights, \(b_k\) is a threshold parameter, \(x_j\) is the input vector and \(y_k\) is the output of the \(k\)th neuron.

![Single Neuron, Multilayer Neural Network](image)

Figure 3. (left side) Single Neuron, (right side) Multilayer Neural Network

Arrangements of such units form the ANN that are characterized by:

- Very simple neuron-like processing elements;
- Weighted connections between the processing elements;
- Highly parallel processing and distributed control;
- Automatic learning of internal representations.

ANN aim to explore the massively parallel network of simple elements in order to yield a result in a very short time slice and, at the same time, with insensitivity to loss and failure of some of the elements of the network. These properties make artificial neural networks appropriate for application in pattern recognition, signal processing, image processing, financing, computer vision, engineering, etc.
There are different architectures of ANN that are dependent upon the learning strategy adopted. This paper briefly describes the Multilayer Perceptron (MLP) with error backpropagation learning. Detailed introductions on ANN can be found in [20] and [21]. MLP with backpropagation learning algorithm, are feedforward networks composed of an input layer, an output layer, and a number of hidden layers, whose aim is to extract high order statistics from the input data [20]. Figure 1(right side) depicts a multilayer neural network with a hidden layer. Functions $\varphi(v)$ provide the activation for the neuron. Neural networks will solve nonlinear problems, if nonlinear activation functions are used for the hidden and/or the output layers. From several activation functions, the sigmoid are commonly used:

$$
\text{bipolar function } \varphi(v) = \frac{1 - \exp(-av)}{1 + \exp(-av)} \quad (24)
$$

A feedforward network is a non-linear mapping to compute the output vector from an input vector. The connections among the several neurons (Fig. 1(b)) have associated weights that are adjusted during the learning process, thus changing the performance of the network. Two distinct phases can be devised while using ANN: the training phase (learning process) and the run phase (activation of the network). The training phase consists of adjusting the weights for the best performance of the network in establishing the mapping of many input/output vector pairs. Once trained, the weights are fixed and the network can be presented to new inputs for which it calculates the corresponding outputs, based on what it has learned.

4. Numerical results

The Lorenz systems was integrated using a second order Runge Kutta methods, with $\Delta t = 10^{-3}$, with initial condition for the Lorenz system: $w_0 = [x_0 y_0 z_0]^T = [1 : 5088701 : 53127125 : 46091]^T$. For training data set, 2000 data are considered, and 333 data are used for cross validation. In order to test multilayer perceptron ANN architectures, for emulating a variational method in data assimilation, momentum constant we used for hidden layer $\alpha_h$ and $\alpha_n(n = x; y; z)$ for the output layer. A similar feature is employed for learning rates: $\eta_h$ for hidden layer, and $\eta_n(n = x; y; z)$ for the output layer. The numerical values for these parameters are shown in table 1.

| Table 1. Neural network parameters: one hidden layer |
|-----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| neurons         | $\alpha_h$    | $\alpha_x$    | $\alpha_y$    | $\alpha_z$    | $\eta_h$      | $\eta_x$      | $\eta_y$      | $\eta_z$      |
|-----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| 2               | 0.6            | 0.6            | 0.6            | 0.6            | 0.001          | 0.001          | 0.001          | 0.001          |

The application of this technique uses the variational functional 6, called objective function, containing the term observation $J_o$ and the term modeling $J_h$. In this work we used only the term of observation $J_o$, ie the functional form:

$$
J(x(t_0)) = \frac{1}{2} \left\{ \frac{1}{N} \sum_{i=0}^{N} (y_i^o - H(x_i))^T R^{-1} (y_i^o - H(x_i)) \right\}^{J_o}_1
$$

where $R$ is the covariance matrix of observation error, $y$ the vector of observations, $H = I$ and $x$ is the vector of state variable dynamical system. The method determines the best fit between the model data and observation data for the period of assimilation. Figure 4 shows the results of assimilation with this technique for the components $x$ and $z$, respectively. The observations were inserted at every 12 time steps. Figures 5 and 6 show the results with the assimilation...
Figure 4. (left side) Time serie of $x$ component; (right side) Time serie of $z$ component. Data assimilation every 12 time steps. Blue curve: truth; green squares: observations; red curve: estimate and black curve: background. Estimated by variational method.

Figure 5. (left side) Time serie of $x$ component; (right side) Time serie of $z$ component. Data assimilation every 500 time steps. Blue curve: truth; green squares: observations; red curve: estimate and black curve: background. Estimated by variational method.

performed every 500 time steps. In Figures 4 and 5, the assimilation window corresponding to timestep 3000 thereafter we have the forecast. Note that for the assimilation window with observations inserted every 12 time steps (Figure 4), it was obtained a good prediction around 12000 time steps, while for the window of assimilation with observations every 500 time steps, we have obtained good prediction up to 10000 time steps. This indicates that the frequency of observations is relevant to the quality of the forecast. In Table 2, the errors for data assimilation process are shown for three frequencies of observations. The artificial neural network was efficient for emulate the variational method, as we can see by the graph of the error in Figure 7, which shows the absolute difference between truth and estimated, and Figure 8 show this error scaled in semi-log.

5. Conclusion
Neural network can be successfully used as a data assimilation method. The Multilayer Perceptron technique was tested in nonlinear systems described by Lorenz equations, which have chaotic dynamics. As neural network employed supervised training, where the target of
Figure 6. (left side) Time serie of $x$ component; (right side) Time serie of $z$ component. Data assimilation every 500 time steps. Blue curve: truth; green curve: estimated. Estimated by multilayer perceptron (MP).

Figure 7. Graph of the error of $x$ components Lorenz’s model for assimilation performed every 12 time steps, (left side): variational method; (right side): multilayer perceptron.

Table 2. Errors: Variational Method and Multilayer Perceptron.

| freq | $x$     | $y$     | $z$     | $(x + y + z)/3$ | $x$     | $y$     | $z$     | $(x + y + z)/3$ |
|------|---------|---------|---------|----------------|---------|---------|---------|----------------|
| 25   | 0.0011  | 0.0015  | 0.0018  | 0.0015         | 0.3186  | 0.3403  | 0.3251  | 0.3280         |
| 50   | 0.0017  | 0.0024  | 0.0029  | 0.0070         | 0.3087  | 0.3450  | 0.3118  | 0.3218         |
| 100  | 0.0020  | 0.0025  | 0.0030  | 0.0025         | 0.4106  | 0.5609  | 0.4937  | 0.4884         |
| 500  | 0.0061  | 0.0085  | 0.0113  | 0.0086         | 0.4341  | 0.6385  | 0.7928  | 0.6218         |

The network was determined by a variational method. This ANN presents a lower computational complexity than the Particle Filter, and variational approach [22] or Kalman filter [16].
Figure 8. Errors at log scale: (left side): variational method; (right side): multilayer perceptron.

References
[1] Daley D 1993 Atmospheric Data Analysis (Cambridge University Press).
[2] Kalnay E 2003 Atmospheric modeling, data assimilation and predictability (Cambridge University Press).
[3] Bennett A F 2002 Inverse Modeling of the Ocean and Atmosphere (Cambridge University Press).
[4] Zannetti P 1990 Air pollution modeling: theories, computational methods and available software (Computational Mechanics Publications)
[5] Garner T W, Wolf R A, Spiro R W, Thomsen M F 1999 First attempt at assimilation data to constrain a magnetospheric model Journal of Geophysical Research 104, 25145-25152.
[6] Schunk R W, Scherliess L, Sojka J J, Thompson D C, Anderson D N, Codrescu M, Minter C, Fuller-Rowell T J, Heelan R A, Hairston M, Howe B M 2004 Global assimilation of ionospheric measurements (GAIM00). Radio Science, 1, RS1052.
[7] Jazwinski A H 1970 Stochastic Processes and Filtering Theory (Academic Press).
[8] Evensen G 2007 The Ensemble Kalman Filter (Springer Verlag).
[9] Belyaev K, Meyers S, O’Brien J J 2000 Fokker-Planck equation application to data assimilation into hydrodynamic models J. Math. Sci., 99(4) 1393-1403.
[10] Belyaev K P, Tanajura C A S, O’Brien J J 2001 A data assimilation method used with an ocean circulation model and its application to the tropical Atlantic Appl. Math. Modelling 25 655-670.
[11] Campos Velho H F, Cintra R S, Furtado H C M. Introdução a assimilação de dados 2007. Available: www.lac.inpe.br/ haroldo/Curso-DataAssimilacao/CursoAssimilacao – INPE – 2.pdf. Accessed: September 22, 2010.
[12] Furtado H C M, Campos Velho H F, Macau E E N “Redes neurais e diferentes métodos de assimilação de dados em dinâmica não linear.” 125 p. (INPE-15235-TDI/1322). Dissertação (Mestrado em Computação Aplicada) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos. 2008. Available: http://urilb.net/sid.inpe.br/mtc-md17080/2008/02.07.10.49.
[13] Nakano S, Ueno G, Higuchi T 2007 Merging particle filter for sequential data assimilation Nonlinear Processes in Geophysics 13 395-408.
[14] Nowosad A G, Rios Neto A, Campos Velho H F 2000 Data assimilation in chaotic dynamics using neural networks. In: International Conference on Nonlinear Dynamics, Chaos, Control and Their Applications in Engineering Sciences 6 212-221 (Associação Brasileira de Ciências Mecânicas).
[15] Härter F P, Campos Velho H F 2005 Recurrent and feedforward neural networks trained with cross correlation applied to the data assimilation in chaotic dynamic Brazilian Journal of Meteorology 20 411-420.
[16] Härter F P, Campos Velho H F 2008 New approach to applying neural network in nonlinear dynamic model Applied Mathematical Modelling 32 2621-2633 - DOI 10.1016/j.apm.2007.09.006 - ISSN: 0307-904X.
[17] Härter F P, Rempel E L, Campos Velho H F, Chian A 2008 Application of artificial neural networks in auroral data assimilation. Journal of Atmospheric and Solar – Terrestrial Physics, 70 1243-1250, - DOI 10.1016/j.jastp.2008.03.018.
[18] Lorenz E N 1963 Deterministic Nonperiodic Flow Journal of the Atmospheric Sciences 20(2) 130-141.
[19] Lorenz E N 1965 A study oh the predictability of a 28-variable model Journal Tellus 17
[20] Haykin S 1993 Neural Networks: A Comprehensive Foundation (New York: Prentice Hall).
[21] Nadler M and Smith E P 1993 Pattern Recognition Engineering (New York: John Wiley & Sons)
[22] Furtado H C M, Campos Velho H F, Macau E E N, Härter F P 2007 Diferent data assimilation methods applied to the Lorenz dynamical system 6th Brazilian Conference on Dynamics, Control and Their Applications (DINCON-2007) May 22-25 (UNESP São José do Rio Preto (SP) Brazil) Proceedings in CD-Rom 1030-1035 - in Portuguese.