CrossMark

Nonlinear regression in environmental sciences using extreme learning machines: A comparative evaluation

Aranildo R. Lima [a,*], Alex J. Cannon [b,1], William W. Hsieh [a]

[a] Department of Earth, Ocean and Atmospheric Sciences, University of British Columbia, Vancouver, BC, Canada
[b] Pacific Climate Impacts Consortium, University of Victoria, Victoria, BC, Canada

ABSTRACT

The extreme learning machine (ELM), a single-hidden layer feedforward neural network algorithm, was tested on nine environmental regression problems. The prediction accuracy and computational speed of the ensemble ELM were evaluated against multiple linear regression (MLR) and three nonlinear machine learning (ML) techniques — artificial neural network (ANN), support vector regression and random forest (RF). Simple automated algorithms were used to estimate the parameters (e.g. number of hidden neurons) needed for model training. Scaling the range of the random weights in ELM improved its performance. Excluding large datasets (with large number of cases and predictors), ELM tended to be the fastest among the nonlinear models. For large datasets, RF tended to be the fastest. ANN and ELM had similar skills, but ELM was much faster than ANN except for large datasets. Generally, the tested ML techniques outperformed MLR, but no single method was best for all the nine datasets.

1. Introduction

Linear models are simple, fast, and often provide adequate and interpretable descriptions of how the predictors affect the outputs. In particular, for prediction purposes they sometimes outperform fancier nonlinear models (Hastie et al., 2009). However, environmental problems are generally complex with many components, such as trends, seasonality, interactions between predictors, and nonlinear relationships (Kuligowski and Barros, 1998; Cawley et al., 2007). Predictions by linear models may have accuracy limitations due to the generalization (and extrapolation) of the environmental problems by linear functions.

To overcome the limitations of linear models, nonlinear machine learning (ML) methods have been successfully used in environmental problems (Cherkassky et al., 2006). Artificial neural networks (ANN), support vector regression (SVR) and random forests (RF) have been used to solve hydrological problems, such as the prediction of 10-day reservoir inflows, downscaling precipitation,

and prediction of water resource variables (e.g. flow, water level, nitrate, salinity and suspended sediment concentration) (Tripathi et al., 2006; Chen et al., 2010; Maier et al., 2010; Cannon, 2012b; Rasouli et al., 2012). Due to the large amount of research activity in hydrology using ANN models, good review papers (Maier and Dandy, 2000; Abrahart et al., 2012; Maier et al., 2010) are available in the hydrological literature. For hydrological problems, gradient-based optimization methods are widely used to train ANN models (Maier et al., 2010). This is also true for environmental problems such as forecasting wind power (Kusiak et al., 2009) and equatorial Pacific sea surface temperatures (Aguilar-Martinez and Hsieh, 2009). However, nonlinear optimization by gradient descent-based learning methods is computationally expensive and may easily converge to local minima. Nature inspired evolutionary computation algorithms have been successfully applied to ANN training (Leung et al., 2003; Chen and Chang, 2009), however there is no consensus on their superior skills or convergence speed (Solomatine and Ostfeld, 2008; Piotrowski and Napiorkowski, 2011). Consequently, there is still a need for better and faster ANN training algorithms.

The extreme learning machine (ELM) algorithm for single-hidden layer feedforward neural networks (SLFNs) was proposed by Huang et al. (2006). The ELM algorithm implements an SLFN similar in structure to a traditional ANN model, but the ELM randomly chooses the weights leading to the hidden nodes or

neurons (HN) and analytically determines the weights at the output layer (Schmidt et al., 1992). Once the activation function has been chosen, the only parameter to be tuned in the ELM is the number of HN. ELM has been used in different research areas (Sun et al., 2008; Huang et al., 2011) and has been found to produce good generalization performance with learning times that are thousands of times faster than traditional gradient-based ANN training algorithms.

Correct adjustment of the model parameters is mandatory for building a successful predictive model. For example, in ANN models, the number of HN, the regularization parameter (i.e. weight penalty parameter), and the algorithm for network training all need to be specified (Haykin, 1998; Hsieh, 2009). Unfortunately (with very few exceptions), there is no unique formula to estimate the model parameters before training starts, so they are usually determined by time-consuming trial and error. In addition, many published articles do not reveal the details of parameter estimation, hiding pitfalls and misuses of the technique employed in the studies (Zhang, 2007; Maier et al., 2010; Wu et al., 2014). For ELM the correct adjustment of the parameter is also crucial. Similar to ANN, the optimal number of HN is problem dependent and unknown in advance (Huang et al., 2006; Guorui et al., 2009; Liu and Wang, 2010). Thus, the process to find the optimal number of HN is also by the trial and error procedure, necessitating multiple ELM runs (Parviainen and Riihimaki, 2013).

The objective of this study is to test whether the claims of ELM having good generalization performance, requiring less computational time than other common nonlinear ML methods and having only one parameter (the number of HN) to adjust, are applicable in the context of environmental sciences. We applied ELM to nine very different datasets. Multiple linear regression (MLR) (with stepwise selection of predictors) was used as the benchmark to compare accuracy (based on the root mean squared error and mean absolute error) and computational time. We also used three nonlinear techniques that have previously been successfully applied to environmental problems as ML benchmarks, namely gradient-based ANN (Cannon and Lord, 2000; Schoof and Pryor, 2001; Krasnopolsky, 2007), SVR (Tripathi et al., 2006; Lima et al., 2013), and RF (Ibarra-Berastegi et al., 2011). We combined the ML techniques with simple automated procedures to search for optimal parameters.

Section 2 describes the datasets used in our study. The regression methods are presented in Sections 3, 4 and 5. Results and discussion of the experiments are given in Section 6, followed by summary and conclusion in Section 7.

2. Data description

In our present peer-reviewed publication system, publications biased in favour of new methods are common because authors tend to present their new methods in the best possible light to enhance their chances of passing the peer review. For instance, an author might test their new method against a traditional method on two different datasets. If the new method fails against a traditional method on one of the datasets, the author could ignore the failure and write a journal paper describing the new method using only the successful case (Hsieh, 2009; Zhang, 2007; Elshorbagy et al., 2010a). Environmental datasets span a broad range in terms of size, number of predictors, degree of nonlinearity, signal-to-noise ratio, etc. Cherkassky et al. (2006) grouped the environmental sciences applications in three domains: climate, Earth and ocean (which is closely related to climate because global processes on the Earth and in the ocean directly influence the climate), and hydrology. Thus, to explore the applicability of the ELM as a prediction tool in environmental sciences, we selected nine different datasets

from the three domains, with four of them in climate (ENSO, YVR, PRECIP and TEMP), two in Earth and ocean (WIND and SO_2), and three in hydrology (SFBAY, FRASER and STAVE) (see the Appendix for details). While the selected datasets by no means span the whole range of environmental datasets, they do provide better insight into the advantages and disadvantages of various methods than a single dataset.

As part of the ML development process, the available data are generally divided into training and testing subsets. The training set is used to build the ML models and the testing set is used to determine the generalization ability of the trained models. Table 1 shows the specifications of the datasets and the basic descriptive statistics of the predictand (i.e. response variable) of the training and testing sets.

One of the most important steps in the ML modelling process is the determination of an appropriate set of predictors (Cherkassky et al., 2006; Solomatine and Ostfeld, 2008; Maier et al., 2010). Usually, they are determined on an ad hoc basis or by using a priori system knowledge (Maier and Dandy, 2000). Since the focus of this research is the comparison of ML models, emphasis was not given to the identification of the optimal predictors for each particular dataset.

3. Extreme learning machines

The key element of an ANN is its distributed, nonlinear structure. An ANN is composed of a large number of highly interconnected neurons (or perceptrons) divided in layers (input, hidden, output), but working in concert to solve a prediction problem (Haykin, 1998).

The multilayer perceptron (MLP) architecture is probably the most popular type of ANN (Kuligowski and Barros, 1998; Cawley et al., 2003; Haylock et al., 2006) and consists of at least one hidden layer sandwiched between the input and output layers. Training of the ANN model involves adjusting the parameters iteratively so the error between the model output $\widehat{\mathbf{y}}$ and the predictand data $\mathbf{y}$ is minimized. The backpropagation algorithm is often used to calculate the gradient of the error function, with a gradient-descent approach used to reduce the mean squared error iteratively, which could be time consuming. There are also others issues to consider such as the number of learning epochs, learning rate, stopping criteria, regularization and/or local minima. To overcome some of these issues, an algorithm called extreme learning machine (ELM) for SLFNs can be used.

The SLFN can be defined as:

$$\widehat{\mathbf{y}}_j = \sum_{i=1}^{L} \beta_i s(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) + \beta_0, \quad (j = 1, \ldots, N) \tag{1}$$

Table 1
Specification of the tested datasets.

Datasets	# observations		# predictors	Predictand			
				Mean		Std. dev.	
	Train	Test		Train	Test	Train	Test
ENSO	264	120	8	−0.049	−0.025	0.902	0.800
SFBAY	475	157	6	1.177	1.145	1.226	1.359
WIND	279	100	160	36,952	40,655	28,610	29,165
FRASER	2557	1095	4	1.779	1.693	0.547	0.528
YVR	3286	1694	3	1.451	1.444	0.457	0.460
STAVE	5258	1260	25	3.134	3.088	0.909	0.919
PRECIP	4213	2074	106	0.984	0.981	0.342	0.341
TEMP	7117	3558	106	−0.002	−0.010	1.010	0.983
SO_2	15,110	7533	27	3.125	3.147	0.929	0.924