Available online at www.isas.org.in/jisas
JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 68(3) 2014 359-364

Nonlinear Support Vector Regression Methodology for Modelling and Prediction: An Application

M.A. Iquebal, Prajneshu and Sarika
Indian Agricultural Statistics Research Institute, New Delhi

Received 05 November 2012; Revised 28 May 2014; Accepted 27 June 2014

SUMMARY

The main limitation of Multiple linear regression analysis for estimating cause-effect relationship is highlighted. Artificial neural network (ANN) methodology that does not require specification of exact nonlinear functional relationship between a response and a set of predictor variables is briefly discussed. Some advantages and disadvantages of this technique are pointed out. The recently developed Nonlinear support vector regression (NLSVR) methodology, which is very promising and versatile, is described. As an illustration, Maize crop yield data as response variable and Total human labour, Farm power, Fertilizer consumption and Pesticide consumption as predictor variables are considered. Both ANN and NLSVR techniques for modelling and prediction purposes are employed. Performance of a fitted model is assessed in terms of Root mean square error (RMSE), Mean absolute error (MAE) and Mean absolute prediction error (MAPE). STATISTICA software package is used for carrying out data analysis. Superiority of NLSVR technique over ANN technique is showed for the data under consideration. It is concluded that NLSVR methodology is quite successful for modelling as well as prediction purposes.

Keywords: Kernel function, Maize crop yield, Mean absolute prediction error, Multilayer perceptron, Nonlinear support vector regression, Polynomial, Radial basis function, Sigmoid.

1. INTRODUCTION

Multiple linear regression (MLR) analysis has been widely used for estimating cause-effect relationship between a response and a set of explanatory variables. However, in reality, the assumption of linearity amongst these variables is rarely satisfied. Further, it is generally not possible to have an idea of the underlying nonlinear relationship. Accordingly, Artificial neural network (ANN) methodology, which is a nonparametric technique, was developed (See *e.g.*, Singh and Prajneshu 2008). A simple ANN model consists of three layers of nodes, viz. Input, hidden, and output layers and allows connection of each node in one layer with every other node in the next layer. Training of network is done by Back-propagation algorithm for estimating the functional relationship between inputs and outputs using supervised learning and by means of adjusting/estimating the weights (strength of connections between the nodes) associated between the nodes at all iterations in order to minimize the sum of squared errors. The net input into a node is given by

$$Netinput_i = \Sigma(w_{ij} * output_j) + u_i, \qquad (1)$$

where w_{ij} are weights connecting neuron j to neuron i; $output_j$ is the output from the unit j and u_i is a threshold for neuron i. Each unit takes its net input and applies an activation function to it. For example, suppose output of the j^{th} unit $g(\Sigma w_{ij} x_i)$, where $g(\cdot)$ is activation function, and x_i is output of the i^{th} unit connected to unit j. The important activation functions generally used are: Identity, tanh, logistic, exponential, and sine.

Corresponding author: Prajneshu
E-mail address : prajneshu@yahoo.co.in

The most popular form of ANN architecture is the Multilayer perceptron (MLP). Typically, it consists of a set of source nodes that constitute the input layer, one or more hidden layers of computation nodes and an output layer of computation nodes. The input signal propagates through the network in a forward direction on a layer-by-layer basis. Another quite popular ANN architecture is Radial basis function (RBF) (Cheng and Titterington 1994), which has a very strong mathematical foundation rooted in regularization theory for solving ill-conditioned problems. An RBF, almost invariably, consists of three layers: A transparent input layer, a hidden layer with sufficiently large number of nodes and an output layer. As its name implies, radially symmetric basis function is used as activation function of hidden nodes. Learning or training is used to describe the process of finding values of the weights w_{ij}. A learning algorithm adjusts connection weights until the system converges to approximately reproduce the output. The optimal weights may be obtained by using Gradient descent algorithm (GDA), Broyden-Fletcher-Goldfarb-Shanno (BFGS), or Conjugate gradient descent algorithm (CGDA) with a view to minimizing sum of the squared error function of the network output (Cheng and Titterington 1994).

Although ANN models have provided a lot of vital information in a large number of instances, they suffer from difficulties with respect to generalization by the possible existence of many local minima or overfitting of data. This is a consequence of the optimization algorithms used for parameter selection and statistical measures to select 'best' model. Nonlinear support vector machine (NLSVM) was developed by V. Vapnik in 1995 at AT&T Bell Laboratories. This is a powerful methodology for solving problems in Nonlinear classification, function estimation and density estimation. In this method, data are mapped into a higher dimensional input space and an optimal separating hyperplane in this space is constructed. The formulation embodies Structural risk minimization (SRM) principle, which has been shown to be superior to traditional Empirical risk minimization (ERM) principle, employed by ANN models. SRM minimizes an upper bound on the expected risk, as opposed to ERM that minimizes the error on training data. This difference equips NLSVM with a greater ability to generalize than ANN. It may be pointed out that NLSVM basically involves solving a convex quadratic programming (QP) problem (Vapnik 2000).

The organization of present article is as follows. After the section on Introduction, various aspects of NLSVR methodology are described in Section 2. Next section, viz. Section 3 deals with an illustration of the methodology to real data. This is followed by some Concluding remarks in last section.

2. NONLINEAR SUPPORT VECTOR REGRESSION (NLSVR) METHODOLOGY

Basically, NLSVM was developed to solve classification problem, but recently it has been extended to the domain of regression problems with the introduction of Vapnik's ε-loss function (Vapnik 2000), and is called as NLSVR. The NLSVR problem generalization is obtained by minimization of the weight vector, which is mainly based on ε-loss function that ignores errors that are within a certain distance ε of the true value and only the points outside ε-tube are penalized. Lin *et al.* (2006) proposed a new approach for NLSVR by giving different weights to different history data points and solving the objective function of QP problem with adjustable punishing coefficient C and Vapnik's loss function ε. For simulated data, it was shown that the proposed approach enhances prediction accuracy. Radhika and Shashi (2009) applied NLSVR methodology to predict daily maximum atmospheric temperature based on previous day's temperature. An excellent review of various aspects of NLSVM is given by Ivanciuc (2007).

Just like Multiple linear regression, basic idea in NLSVR is to find a function that approximates training points well by minimizing prediction error. But the major difference is that all deviations up to a user-specified parameter ε are simply discarded. Also, while minimizing the error, risk of overfitting is reduced by simultaneously trying to maximize flatness of the function. Another difference is that what is minimized is normally the predictions' absolute error instead of the squared error used in Multiple linear regression analysis. Consider a training dataset $g = \{(x_1, y_1), (x_2, y_2), ..., (x_p, y_p)\}$, such that $x_i \in \Re^n$ is a vector of input variables and $y_i \in \Re$ is the corresponding scalar output (target) value. Here, modelling objective is to find a regression function $\boldsymbol{y} = f(\boldsymbol{x})$, such that it accurately predicts output $\{\boldsymbol{y}\}$ corresponding to a new set of input-output data, $\{(\boldsymbol{x}, \boldsymbol{y})\}$, which is drawn from the same underlying joint probability distribution as training set. To fulfil the stated goal, NLSVR considers following estimation function:

$$f(\boldsymbol{x}) = <\boldsymbol{w}, \boldsymbol{x}> + b, \quad (2)$$

where $\boldsymbol{w}$ denotes the weight vector; b refers to a constant known as "bias", $f(\boldsymbol{x})$ denotes a function termed feature, and $<\boldsymbol{w}, \boldsymbol{x}>$ represents the dot product in the feature space, l, such that $\psi:\boldsymbol{x} \rightarrow l$, $\boldsymbol{w} \in l$. The basic concept of NLSVR is to map nonlinearly original data $\boldsymbol{x}$ into a higher dimensional feature space and solve the linear regression problem in this feature space. The regression problem is equivalent to minimization of the following regularized risk function:

$$R(f) = \frac{1}{n}\sum_{i=1}^{n} L(f(x_i) - y_i) + \frac{1}{2}\|\boldsymbol{w}\|^2 \quad (3)$$

where

$$L(f(\boldsymbol{x}) - y) = \begin{cases} |f(\boldsymbol{x}) - y| - \varepsilon, & |f(\boldsymbol{x}) - y| \geq \varepsilon \\ 0, & otherwise, \end{cases} \quad (4)$$

is called the ε-insensitive loss function. The precision parameter represents radius of tube located around the regression function, as shown in Fig. 1. Substituting Eq.(4) in Eq.(3), the optimization problem becomes:

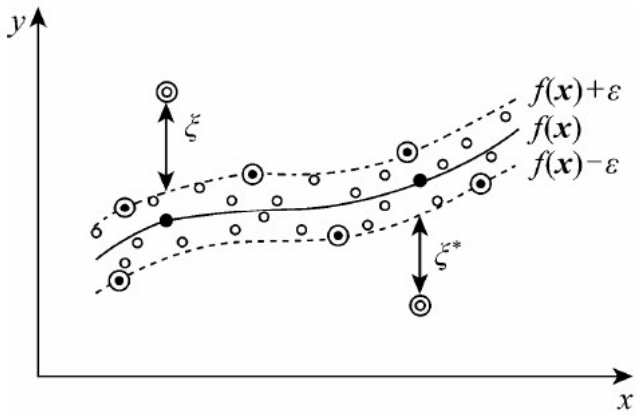

Fig. 1. A schematic diagram of support vector regression using ε-loss function

$$\text{Minimize } \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \quad (5)$$

subject to

$$\begin{cases} y_i - <\boldsymbol{w}, x_i> - b \leq \varepsilon + \xi_i \\ <\boldsymbol{w}, x_i> + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0, \end{cases} \quad (6)$$

where constant $C > 0$ stands for penalty degree of the sample with error exceeding ε. Two positive slack variables ξ_i, ξ_i^* represent the distance from actual values to corresponding boundary values of ε-tube. The dual problem can then be derived by using optimization method to maximize the function:

$$\text{Maximize } -\frac{1}{2}\sum_{i,\,j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)(x_i, x_j) - \varepsilon\sum_{i=1}^{n}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*) \quad (7)$$

subject to

$$\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \text{ and } 0 \leq \alpha_i, \alpha_i^* \leq C,$$

where α_i, α_i^* are Lagrange multipliers. Owing to the specific character of above described QP problem, only some of the coefficients $(\alpha_i^* - \alpha_i)$ are non-zero and corresponding input vectors, x_i are called Support vectors (SVs). These can be thought of as the most informative data points that compress the information content of training set. The coefficients α_i and α_i^* have an intuitive interpretation as forces pushing and pulling the regression estimate $f(x_i)$ towards the measurements y_i. The function obtained by fitting above mentioned maximization function is given by

$$f(\boldsymbol{x}) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) < x_i, x_j > + b. \quad (8)$$

2.1 Kernel Tricks

Let $\boldsymbol{x}$ be a vector in the n-dimensional input space and $\Phi(\cdot)$ be a nonlinear mapping function from input space to high-dimensional feature space, which can be of infinite dimension. Then kernel function can be expressed as the inner product (often called the dot product) and is given by

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i).\ \Phi(\boldsymbol{x}_j)$$

where

$$\Phi(\boldsymbol{x}) = \sqrt{\lambda_i}\Phi(\boldsymbol{x}_i)$$

and λ_i is a positive number. The application of $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is called the *kernel trick*, which enables us to work in a huge dimensional feature space without actual explicit computation in this space. Computations are done in another space after applying this kernel trick. So, an advantage of NLSVR technique is that the nonlinear function $\Phi(\boldsymbol{x}_i)$ need not be used. The most widely known kernel functions used in practice are: Polynomial, Radial basis function (RBF), and Sigmoid, which are defined as follows (Cristianini and Shawe-Taylor 2000):

(a) Polynomial of degree d

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\lambda \boldsymbol{x}_i' \boldsymbol{x}_j + r)^d$$

(b) Radial basis function (RBF)

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = exp\{-\lambda ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2/\sigma^2\}$$

(c) Sigmoid

$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = tanh\{\lambda \boldsymbol{x}_i' \boldsymbol{x}_j + r\}$, where tanh represents hyperbolic tangent function.

Depending on the kernel type chosen, kernel parameters (λ, r, d) have to be set. Choice of best kernel type (and parameters) depends on the application and can be determined by using cross-validation. As a rule of thumb, if the number of features is large (as in the field of text classification), linear kernel is sufficient.

2.2 K-fold Cross Validation

Cross validation (CV) is a standard technique for adjusting hyperparameters of predictive models. In K-fold CV, available data S is partitioned into K subsets $S_1, S_2, ..., S_K$. Each data point in S is randomly assigned to one of the subsets such that these are of almost equal size, *i.e.*, $[|S|/K] \leq S_i \leq [|S|/K]$. Further, define $\overline{S_i} = U_{j=1,...,K \wedge j \neq i} S_i$ as the union of all data points except those in S_i. For each $i = 1, 2, ..., K$, an individual model is built by applying the algorithm to training data S_i. This model is then evaluated by means of a cost function using test data in S_i. Average of K outcomes of the model evaluations is called cross validation (test) performance or cross validation (test) error and is used as a predictor of performance of the algorithm when applied to S. The widely used values for K are 5 and 10 (Hastie *et al.* 2009).

3. AN ILLUSTRATION

As an illustration, data from Singh *et al.* (2004) covering 170 farmers from the state of Uttar Pradesh, India are considered. Specifically, the response variable is 'Maize crop yield (Quintals/hectare)' and four predictor variables are Total human labour (Rupees/hectare), Farm power (Rupees/hectare), Fertilizer consumption (Kilogram/hectare), and Pesticide consumption (Rupees/hectare). Summary statistics for the response and predictor variables from the 170 farmers are given in Table 1.

Table 1. Summary statistics of the response and predictor variables for 170 farmers

Variable*	Mean	Standard Deviation	Range	Coefficient of Variation	Lower Quartile	Upper Quartile
Yield	29.5	10.9	40.9	37.0	18.8	37.4
THL	4256.3	1933.0	11521.6	45.4	2723.5	5480.9
FP	1339.9	611.1	3582.3	45.6	865.4	1725.0
FC	158.9	42.5	355.6	26.7	137.3	175.0
PC	368.7	251.6	1575.0	68.2	186.6	500.0

*Yield: Maize crop yield, THL: Total human labour, FP: Farm power, FC: Fertilizer consumption, and PC: Pesticide consumption

It may be mentioned that entire data analysis is carried out using STATISTICA, Ver. 6.0 software package (2001). In the first instance, attempt is made to fit ANN models to data. Before training, available 170 observations are divided into two subsets: (i) first sub-set is training set comprising 160 observations, which is used for computing and updating the network weight and biases, and (ii) test set comprises the remaining 10 observations. The MLP and RBF networks are trained using all the three learning algorithms, viz. GDA, BGFS and CGDA. Several learning rates (Cheng and Titterington 1994) are considered for training the networks as well as for adjusting the weights. A higher learning rate may converge more quickly but may also exhibit greater instability. For our data, best result is obtained for learning rate as 0.1. For hidden units and output units, several activation functions, viz. Identity, tanh, logistic, exponential and sine are tried. Performance of the trained network is assessed by computing Root mean square error (RMSE) and Mean absolute error (MAE) on the training and test sets. The values of these criteria for the best trained MLP (4-8-1) using BFGS training algorithm and logistic activation function for hidden and output units are computed respectively as 19.60 and 17.86.

Subsequently, NLSVR is trained using all three kernel functions, viz. Polynomial of degree 2, Radial basis function (RBF), and Sigmoid function. Stopping criteria used are: Maximum number as 1000, and difference in RMSE values as 0.001. Further, 5-fold cross validation is applied here. Number of support vectors is computed as 49, of which 38 are bounded. The values of parameters C and ε are respectively

obtained as 18 and 0.05. The estimated value for parameter λ is 0.25. Performance of the NLSVR methodology is assessed through RMSE and MAE criteria. RBF is identified as the best kernel function having computed values respectively as 8.85 and 7.39. A perusal of Table 2 clearly shows superiority of NLSVR over ANN methodologies for the data under consideration. In Fig. 2, the dots indicate various predicted values (obtained by fitting NLSVR model with RBF kernel function) against corresponding actual values and the solid line indicates the ideal line on which predicted and actual values are equal. As the dots are generally quite close to the solid line, it implies that the model provides a reasonably good fit to the data.

Table 2. Performance of MLP and NLSVR methods for modelling

Measures	NLSVR			MLP (4-8-1)
	Polynomial of degree 2	RBF	Sigmoid	
RMSE	9.65	8.85	11.25	19.60
MAE	7.80	7.39	8.39	17.86

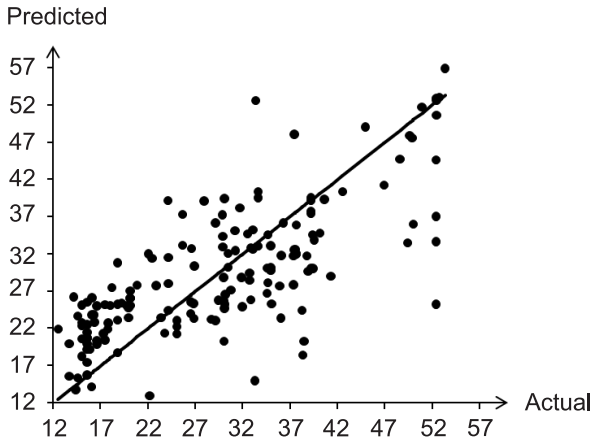

Fig. 2. Graph of predicted and actual maize crop yields (Quintals/ hectare)

Finally, for the test data comprising 10 observations, predicted values along with actual values and goodness-of-fit criteria obtained using fitted MLP (4-8-1) and NLSVR models are reported in Table 3. Evidently, superiority of NLSVR model over ANN model is demonstrated for prediction purpose also. Finally, it may be noted that predicted values by NLSVR model are again quite close to actual values. Thus, it is concluded that nonlinear support vector regression methodology is suitable for describing and prediction purposes for the data under consideration.

Table 3. Predicted maize crop yield and Goodness-of-fit measures for various models

Actual	Predicted values using	
	NLSVR with RBF Kernel function	MLP (4-8-1)
25.00	27.38	20.55
36.14	38.05	38.76
43.67	41.67	40.48
22.32	25.02	21.79
29.94	31.68	32.37
37.31	39.30	41.80
32.93	32.89	36.29
36.32	36.47	34.75
18.75	21.44	18.90
17.75	18.29	14.82
Goodness-of-fit measures:		
RMSE	1.87	2.92
MAE	1.61	2.57
MAPE	6.05	8.66

4. CONCLUDING REMARKS

In this article, potential of Nonlinear support vector regression methodology (NLSVR) is highlighted for tackling the realistic situation in which exact nonlinear functional relationship between response variable and a set of predictors is unknown. Superiority of this approach over Artificial neural network methodology is shown for the data under consideration. Although NLSVR may not be able to provide the same level of insight as many Mechanistic models do, it is not correct to treat them as "black boxes". It is hoped that, in future, research workers would start applying NLSVR methodology extensively for modelling and prediction purposes. Here, three kernel functions, viz. Polynomial, Radial basis function, and Sigmoid are tried. Determination of optimum kernel function for a specific data is a researchable issue. Work is in progress to fit NLSVR model through Particle Swarm optimization procedure (Kennedy and Eberhart 1995) and shall be reported separately in due course of time.

ACKNOWLEDGEMENTS

Authors are grateful to the referee for valuable comments.

REFERENCES

Cheng, B. and Titterington, D.M. (1994). Neural networks: A review from a statistical perspective. *Statist. Sci.*, **9,** 2-54.

Cristianini, N. and Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, U.K.

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. 2nd Ed. Springer-Verlag, New York.

Ivanciuc, O. (2007). Applications of support vector machines in chemistry. In: *Reviews in Computational Chemistry*, 23, Eds.: K.B. Lipkowitz and T.R. Cundari, Wiley-VCH, Weinheim, 291-400.

Kennedy, J. and Eberhart, R.C. (1995). Particle Swarm Optimization. In: *Proceedings of IEEE International Conference on Neural Networks,* IEEE Press, Piscataway, N.J., **IV**, 1942-1948.

Lin, J.Y., Cheng, C.T. and Chau, K.W. (2006). Using support vector machines for long-term discharge prediction. *Hydrol. Sci. J.*, **51**, 599-612.

Radhika, Y. and Shashi, M. (2009). Atmospheric temperature prediction using support vector machines. *Intt. J. Commu. Theo. Engg.*, **1**, 55-58.

Singh, R.K. and Prajneshu (2008). Artificial neural network methodology for modelling and forecasting maize crop yield. *Agric. Econ. Res. Rev.*, **21**, 5-10.

Singh, R.P., Kumar, R., Singh, B.B., Awasthi, P.K., Atibudhi, H.N., Chahal, S.S., Varghese, K.A., Singh, R.K. and Maurya, S.P. (2004). Technological Change and Production Performance in Irrigated Maize based Agro-ecosystem: The Interplay of Economic, Technological and Institutional Factors. N.A.T.P. (PSR-61). I.A.R.I. Research Report 2004-01, pp. 1-107, New Delhi.

StatSoft, Inc. (2001). STATISTICA (data analysis software system),Version 6.0. *(www.statsoft.com)*.

Vapnik, V. (2000). *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.