SUPPORT VECTOR REGRESSION METHOD FOR BOUNDARY VALUE PROBLEMS

KUN FU, YOU-HUA WANG, YONG-FENG DONG, XIANG-DAN HOU, XUE-QIN SHEN, WEI-LI YAN

HeBei University of Technology, TianJin 300130, China
E-MAIL: fukun1979316@sohu.com, Dongyf2000@126.com

Abstract:

This article presents a method to solve boundary value problems using support vector regression and radial basis function network. The boundary is determined by a number of points that belong to it and are closely located, so as to offer a reasonable representation. Two methods are employed: a support vector regression is constructed to have part of effect on the boundary conditions as the basic approximate element and contains adjustable parameters; a radial basis function network is used to account for the exact satisfaction of the boundary conditions. The method was used to solve a two-dimensional partial differential equation and had gained feasible accurate result. This method is completely practical in technology.

Keywords:

Support vector regression; support vector machine; boundary value problems; partial differential equation radial basis function network

1. Introduction

In this article a method to deal with boundary value problems with regular or irregular boundaries is proposed. This approach is based on the synergy of two methods [1], [2]: a support vector regression (SVR) as the basic approximating element and a radial basis function (RBF) network used to satisfy the boundary conditions (BC). SVR is a branch of support vector machine (SVM) [3], [4], [5].

2. Description of the method

Partial differential equations (PDE) of the form will be examined:

$$L\psi = f$$

$$\text{Subject to } \psi(\boldsymbol{r}_i) = b_i,\ i=1, 2, \ldots, m \text{ (Dirichlet)} \quad (1)$$

here L is a differential operator and $\psi = \psi(x)$ ($\boldsymbol{x} \in D \subset R^n$) with Dirichlet BC. The boundary B can be any arbitrarily complex geometrical shape. Here the boundary is defined as a set of points that are chosen so as to represent its shape with reasonable accuracy. So the m points $\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{r}_m \in B$ are chosen to represent the boundary.

To obtain a numerical solution to the above differential equation, the method is adopted which assumes the discretization of the domain D into a set of points $\hat{D}$ ($\boldsymbol{x}_i$, i=1, 2, …, l). The problem is then transformed into the following system of equations:

$$L\psi(\boldsymbol{x}_i) = f(\boldsymbol{x}_i),\quad \forall x_i \in \hat{D}, \qquad i=1, 2, \ldots, l$$

$$\text{Subject to } \psi(\boldsymbol{r}_i) = b_i,\quad \forall \boldsymbol{r}_i \in B \qquad i=1, 2, \ldots, m \quad (2)$$

Let $\psi_t(\boldsymbol{x}, p)$ denote a trial solution to the above problem where p stands for a set of model parameters to be adjusted. In this way, the problem is transformed into the following constrained minimization problem:

$$\text{Minimize} \sum_{i=1}^{l} \left[L\psi_t(\boldsymbol{x}_i, p) - f(\boldsymbol{x}_i)\right]^2$$

$$\text{Subject to } \psi(\boldsymbol{r}_i) = b_i,\ i=1, 2, \ldots, m \text{ (Dirichlet)} \quad (3)$$

The constrained optimization problem may be tackled in this way: devise a model, such that the constraints are exactly satisfied by construction and hence use those unconstrained optimization techniques. A model suitable for the approach is based on the synergy of two methods, and it can be written as [1]:

$$\psi_t(\boldsymbol{x}, p) = S(\boldsymbol{x}, p) + \sum_{l=1}^{m} \boldsymbol{q}_l e^{-\lambda|\boldsymbol{x} - \alpha \boldsymbol{r}_l + \boldsymbol{h}|^2} \quad (4)$$

|.| denotes the Euclidean norm ($|a| = (\sum_{i,j}^{n} |a_{ij}|^2)^{\frac{1}{2}}$), and $S(\boldsymbol{x}, p)$ is a SVR with p denoting a set of its parameters. The second term in the right of above formulation (4) represents a RBF [6] network with m hidden units that all share a

common exponential factor λ. Here α is a scalar constant and h is a constant vector. The parameters λ, α and $\boldsymbol{h}$ must be chosen appropriately so as to ease the numerical task.

For a given set p parameters of SVR, the coefficients $\boldsymbol{q}_l$ are uniquely determined by requiring that the boundary conditions are satisfied for i=1, 2, …, m:

$$b_i - S(\boldsymbol{r}_i, p) = \sum_{l=1}^{m} \boldsymbol{q}_l e^{-\lambda|\boldsymbol{r}_i - \alpha \boldsymbol{r}_l + \boldsymbol{h}|^2} \quad \text{(Dirichlet)} \qquad (5)$$

Therefore, in order to obtain the parameters $\boldsymbol{q}$ that satisfy the BC, the following linear system [2] has to be solved:

$$A\boldsymbol{q} = \gamma, A_{i,l} = e^{-\lambda|\boldsymbol{r}_i - \alpha \boldsymbol{r}_l + \boldsymbol{h}|^2}, \gamma_i = b_i - S(\boldsymbol{r}_i, p) \quad \text{(Dirichlet)} \qquad (6)$$

The model based on the SVR-RBF synergy satisfies exactly the BC.

3. SVR Implementation

Below is a brief review for SVR [7], [8]. Given l sample points $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_l, y_l)$ $\boldsymbol{x}_i \in R^n$, $y_i \in R$ (i= 1, 2, … ,l). The objection is to find a regression function $f(\boldsymbol{x})$ by the principle of risk minimization:

$$R[f] = \int c\,(\boldsymbol{x}, y, f(\boldsymbol{x}))dP(\boldsymbol{x}, y) \qquad (7)$$

$R[f]$ is the expected risk, $c\,(\boldsymbol{x}, y, f(\boldsymbol{x}))$ is the lost function, $P(\boldsymbol{x}, y)$ is the $(\boldsymbol{x}, y)$ distributing function. For $P(\boldsymbol{x}, y)$ is unknown commonly, $R[f]$ can be approximate by the following formulation:

$$R[f] \approx R_{emp}[f] + \Phi = \frac{1}{l}\sum_{i=1}^{l} c\,(\boldsymbol{x}_i, y_i, f(\boldsymbol{x}_i)) + \Phi \qquad (8)$$

$R_{emp}[f]$ is the empirical risk, Φ is the structural risk function which relates to VC dimension and by which the regression function performance can be mended. This is the principle of structural risk minimization (SRM).

Now consider the case of linear regression:

$$f(\boldsymbol{x}) = (\boldsymbol{w} \,.\, \boldsymbol{x}) + b \qquad (9)$$

$\boldsymbol{w} \in R^n$, $b \in R$, (.) is the dot products in R^n. The method should try to made regression function f flatter. Formally the above problem can be wrote as a convex optimization problem by requiring:

$$\text{Minimize } \frac{1}{2}\|\boldsymbol{w}\|^2$$

$$\text{Subject to } \begin{cases} y_i - (\boldsymbol{w}.\boldsymbol{x}_i) - b \le \varepsilon \\ (\boldsymbol{w}.\boldsymbol{x}_i) + b - y_i \le \varepsilon \end{cases} \qquad (10)$$

The tacit assumption was that such a function of actually exists that approximates all pairs $(\boldsymbol{x}_i, y_i)$ with ε precision, or in other words, that the convex optimization problem is feasible. Sometimes, however, this may not be the case, or some errors may be also wanted to allow, so the slack variable ξ_i, ξ_i^* to cope with otherwise infeasible constrains of the optimization problem can be introduced. Hence the formulation is as follows:

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

$$\text{Subject to } \begin{cases} y_i - (\boldsymbol{w}.\boldsymbol{x}_i) - b \le \varepsilon + \xi_i \\ y_i + (\boldsymbol{w}.\boldsymbol{x}_i) - b \le \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \ge 0 \end{cases} \qquad (11)$$

The constant $C>0$ determines the trade off between the flatness of f and the amount up to which deviations larger than ε are tolerated.

A Lagrange function minimization problem from the above convex optimization problem is constructed, then the Lagrange function minimization problem can be transformed into a Quadratic Programming (QP) problem:

$$\text{Maximize } W = -\varepsilon\sum_{i=1}^{l}(\alpha_i^* + \alpha_i) + \sum_{i=1}^{l} y_i(\alpha_i^* - \alpha_i) - \frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)(\boldsymbol{x}_i . \boldsymbol{x}_j)$$

$$\text{Subject to } \begin{cases} \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0 \\ \alpha_i, \alpha_i^* \in [0, C] \end{cases} \qquad (12)$$

Here α_i, α_i^* (i= 1, 2, … ,l) are the Lagrange multipliers, by finding an optimal set of Lagrange multipliers, the $\boldsymbol{w}$ is as follows:

$$\boldsymbol{w} = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\boldsymbol{x}_i \qquad (13)$$

Then b can be sought through the KKT conditions, so $f(\boldsymbol{x})$ can be found:

$$f(\boldsymbol{x}) = \sum_{i=1}^{l}(\alpha_i^* - \alpha_i)(\boldsymbol{x}_i . \boldsymbol{x}) + b = \sum_{i \in sv}(\alpha_i^* - \alpha_i)(\boldsymbol{x}_i . \boldsymbol{x}) + b \qquad (14)$$

This is the so-called support vector expansion, and the $\boldsymbol{x}_i$ corresponding to α_i, $\alpha_i^* \neq 0$ is the support vector.

The above problem is the linear SVR. Then the nonlinear SVR can be discussed. As noted already in the previous introduction, the SVR algorithm only depends on dot products between the various patterns. Hence it suffices to know and use $K(\boldsymbol{x}, \boldsymbol{x}') = (\phi(\boldsymbol{x}), \phi(\boldsymbol{x}'))$ instead of $\phi(.)$ explicitly. This allows one to rewrite the $\boldsymbol{w}$ and $f(\boldsymbol{x})$ as follows:

$$\boldsymbol{w} = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)\phi(\boldsymbol{x}_i) \tag{15}$$

$$f(\boldsymbol{x}) = \sum_{i=1}^{l} (\alpha_i^* - \alpha_i)K(\boldsymbol{x}_i, \boldsymbol{x}) + b \tag{16}$$

A version of Platt's sequential minimal optimization (SMO) [9], [10] algorithm for the regression case is to be implemented. SMO is an extreme of the decomposition method where, the working set is restricted to only two elements. The main advantage is that each two-variable sub-problem can be analytically solved.

In order to apply the proposed method in formulation (6), the values of λ, α and $\boldsymbol{h}$ must be specified to define the linear systems (matrices A). In the experiments the linear systems were solved using standard LU decomposition. For the Dirichlet case α=1 and $\boldsymbol{h}$ =0 were adequate to produce a nonsingular well-behaved matrix A. For large values λ, they affect the model only in the neighborhood of the boundary points. In other words, the RBF contributes a correction that accounts for the BC. So λ must be selected with caution. A good choice is found to be: λ=1/d, where d is the minimum distance between any two points on the boundary .Now describe the strategy followed in detail.

First, SVR is used to obtain that $S(\boldsymbol{x}, p)$ approximates the solution both inside the domain and on the boundary. Since this approach is efficient this task complete rather quickly.

Second, RBF method is then started, employing the $S(\boldsymbol{x}, p)$ that was previously obtained. RBF network contributes as a correction to the SVR mainly around the boundary points. Therefore the SVR-RBF method starts from a low error value and requires only a few optimization steps in order to yield a solution to satisfy the BC exactly.

4. Example

Here an example in two dimensions is to be solved. The boundary of it is a square. Here the boundary is defined by a set of points that belong to it, and hence we treat it as such with the method described in the present article.

$$\begin{cases} \dfrac{\partial^2 u(x,y)}{\partial x^2} + \dfrac{\partial^2 u(x,y)}{\partial y^2} \\ = e^{-x}(x-2+y^3+6y), (x,y) \in (0,1)\times(0,1) \\ u(0,y) = y^3 \\ u(1,y) = \dfrac{1+y^3}{e} \\ u(x,0) = xe^{-x} \\ u(x,1) = e^{-x}(1+x) \end{cases} \tag{17}$$

The analytic solution is: $u(x,y) = e^{-x}(x+y^3)$.

According to the above method, the trial form of the solution $S(x,y)$ is taken to be:

$$\begin{aligned} S(x,y) &= x + w_1 y^3 + w_2 x^2 + w_3 xy^3 \\ &+ w_4 x^3 + w_5 x^2 y^3 + b\left(-\frac{x^4}{6} - \frac{x^3 y^3}{6}\right) \end{aligned} \tag{18}$$

Now do the quadratic derivations to the x and y then train the data points with SVR. The points are considered by dividing the interval [0, 1] on the x axis and y axis respectively, using equidistant points. The total number of points taken is 121. The initial parameters are: $\boldsymbol{h}$ =0, C =0.009; by the learning of SVR the w_1=1.00198, w_2=-0.993809, w_3=-1.00815, w_4=0.466713, w_5=0.476908, b=0.626751. Figure 1 displays the results of the analytic solution and regression solution and Figure 2 and Figure 3 are the absolute error and the relative error.

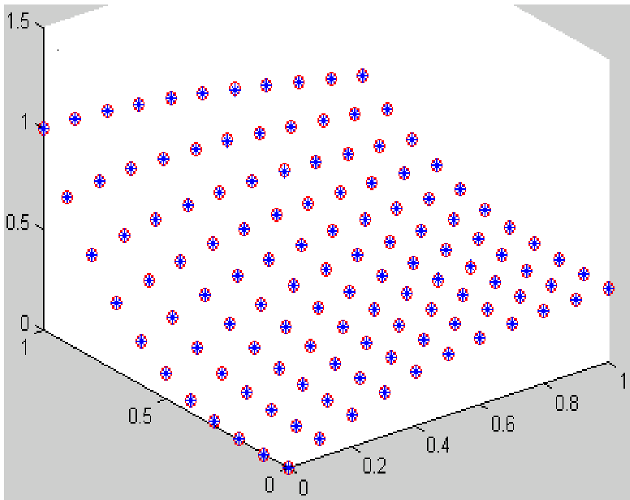

Figure 1. Analytic solution and regression solution

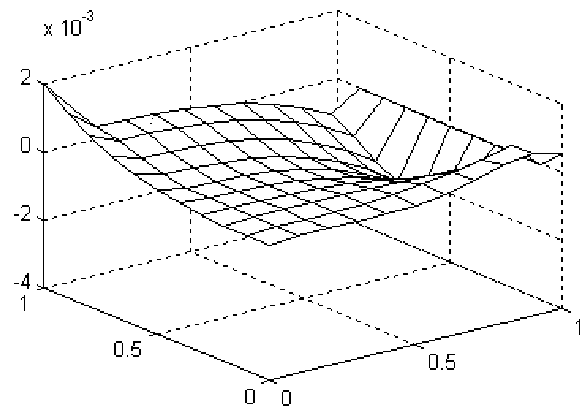

Figure 2. Absolute error

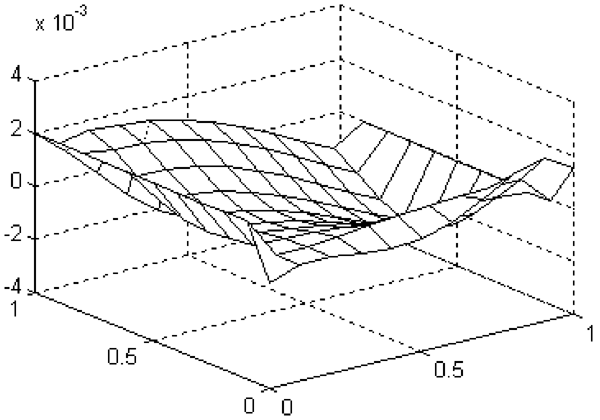

Figure 3. Relative error

5. Conclusion

The method provides accurate solution in a closed analytic form that satisfies the boundary conditions at the selected points exactly, the range of the accuracy is from 10^{-3} to 10^{-6}. So the method is feasible in technology.

Acknowledgements

This work is supported by natural science fund of HeBei Province under grant No. (603073).

References

[1] Isaac Elias Lagaris, and Aristidis Likas, "Neural-Network Methods for Boundary Value Problems with Irregular Boundaries", IEEE Transactions on Neural Network, pp. 1041~1049 September 2000.

[2] A J Meade, and A A Fernandez, "The Numerical Solution of Linear Ordinary Differential Equations by Feed-forward Neural Networks", Math Compute Modeling, pp. 1-25, 1994.

[3] Vapnik V N, Statistical Learning Theory, J.Wiley, New York, 1998.

[4] A J Smola, and B.Schkopf, A Tutorial on Support Vector Regression, http://www.neurocolt.com, 2000.

[5] Chen Paihsuen, Lin Chihjen, and Bernhard Schkopf, A Tutorial on ν-Support Vector Machines, http://www.csie.ntu.edu.tw/~cjlin/papers/nusvmtutorial.pdf, 2003.

[6] Marinaro M, and Scarpetta S, "On-line learning in RBF neural networks: A stochastic approach", Neural Networks, pp. 719-729, September 2000.

[7] Chalimourda Athanassia, Scholkopf Bernhard, and Smola Alex J, " Experimentally Optimal ν in Support Vector Regression for Different Noise Models and Parameter Settings", Neural Networks, pp. 127-141, January, 2004.

[8] Cherkassky Vladimir, and Ma Yunqian, " Practical Selection of SVM Parameters and Noise Estimation for SVM Regression ", Neural Networks, pp. 113-126, January, 2004.

[9] David Andre, "Exploring Online Support Vector Regression", http://www.cs.berkeley.edu/~dandre, 1999.

[10] John C Platt, "Fast Training of Support Vector Machines using Sequential Minimal Optimization", MIT Press, pp. 185-208, 1999.