Expert Systems with Applications 39 (2012) 8220–8227

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

New validation methods for improving standard and multi-parametric support vector regression training time

J. Gascón-Moreno, S. Salcedo-Sanz *, E.G. Ortiz-García, J. Acevedo-Rodríguez, Jose A. Portilla-Figueras

Department of Signal Theory and Communications, Universidad de Alcalá, Madrid, Spain

ARTICLE INFO

Keywords:
Support vector regression algorithms
SVMr hyper-parameters
Training time
Validation methods

ABSTRACT

The selection of hyper-parameters in support vector regression algorithms (SVMr) is an essential process in the training of these learning machines. Unfortunately, there is not an exact method to obtain the optimal values of SVMr hyper-parameters. Therefore, it is necessary to use a search algorithm and sometimes a validation method in order to find the best combination of hyper-parameters. The problem is that the SVMr training time can be huge in large training databases if standard search algorithms and validation methods (such as grid search and *K*-fold cross validation), are used. In this paper we propose two novel validation methods which reduce the SVMr training time, maintaining the accuracy of the final machine. We show the good performance of both methods in the standard SVMr with 3 hyper-parameters (where the hyper-parameters search is usually carried out by means of a grid search) and also in the extension to multi-parametric kernels, where meta-heuristic approaches such as evolutionary algorithms must be used to look for the best set of SVMr hyper-parameters. In all cases the new validation methods have provided very good results in terms of training time, without affecting the final SVMr accuracy.

1. Introduction

The support vector regression algorithm (SVMr) (Smola & Schölkopf, 1998) is a robust methodology in statistical machine learning, that has been successfully applied to solve regression problems (He, Wang, & Jiang, 2008; Lázaro, Santamaría, Pérez-Cruz, & Artés-Rodríguez, 2005; Mohandes, Halawani, Rehman, & Hussain, 2004; Wu, Chau, & Li, 2008). The SVMr uses *kernel theory* (Smola & Schölkopf, 1998) to increase the quality of regression models and, in most cases can be solved as a convex optimization problem. Several fast algorithms can be used to carry out the SVMr training, such as the sequential minimal optimization algorithm (Smola & Schölkopf, 1998). In spite of this, the time for training a SVMr model can be very high due to the SVMr performance heavily depends on the choice of several hyper-parameters, necessary to define the optimization problem and the final SVMr model. There are different approaches focused on reducing this hard computation time of the SVMr model: in Guo and Zhang (2007) a method based on reducing the number of the samples included in the SVMr training is proposed, and in Zhao, Sun, and Zou (2010) a similar idea is applied to multi-parametric kernel SVMr. In Zhao and Sun (2010) a different methodology is applied, based on approximating the SVMr solution instead of solving the optimization problem in an exact way. In Ortiz-Garcia, Salcedo-Sanz, Pérez-Bellido, and Portilla-Figueras (2009) an approach to reduce the SVMr training time based on reducing the hyper-parameters search space is proposed.

The search of the best set of SVMr hyper-parameters is maybe the most time-consuming process in SVMr training: since there is not an exact method to obtain the optimal set of SVMr hyper-parameters, exhaustive search or meta-heuristic based algorithms are usually applied (Akay, 2009; Hou & Li, 2009; Wu, Tzeng, & Lin, 2009). In both cases, two processes must be used to find a good combination of SVMr hyper-parameters: a search algorithm and a validation method. Maybe the most used search algorithm applied to obtain SVMr hyper-parameters is the Grid Search (GS) (Akay, 2009), where the search space of parameters is divided into groups of possible parameters to be tested (usually, an uniform partition of the search space is considered). This algorithm can be easily implemented, however, it has an important drawback: when the number of hyper-parameter combinations is large, the training time becomes high, even considering only the three standard SVMr hyper-parameters in the search, i.e, C, ϵ and γ. In the case of multi-parametric kernel optimization (Friedrichs & Igel, 2005; Zhao & Sun, 2011), with N hyper-parameters to be optimized (C, ϵ, γ_m), the GS approach is computationally not affordable, and meta-heuristic approaches such as evolutionary algorithms (Eiben & Smith, 2003) are usually employed to do this task (Friedrichs & Igel, 2005; Rojas & Fernández-Reyes, 2005).

* Corresponding author. Address: Department of Signal Theory and Communications, Universidad de Alcalá, 28871 Alcalá de Henares, Madrid, Spain. Tel.: +34 91 885 6731; fax: +34 91 885 6699.
E-mail address: sancho.salcedo@uah.es (S. Salcedo-Sanz).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2012.01.142

Independently of the type of algorithm used to carry out the search of the SVMr hyper-parameters, it needs a process to evaluate the goodness of every set of tested parameters (C,ϵ,γ) in the case of the standard SVMr, or (C,ϵ,γ_m) in the case of the multi-parametric SVMr. This is the aim of the *validation method* which is a crucial part of the training process of a SVMr. A given validation method must select the best values for the vector (C,ϵ,γ) or (C,ϵ,γ_m) from the training data. A wrong evaluation at this time could produce over-fitting in the final SVMr model, so its performance in an independent test set will be poor. Most of the authors use traditional validation methods such as *K*-fold cross validation, Leave One Out, Bootstrap, etc., but these validation techniques are not focused on improving the SVMr training time, so it can be really high in applications with large training databases. Note that this high training time is specially important in multi-parametric SVMr, since, as has been mentioned before, the hyper-parameters' search in this approach is usually carried out with evolutionary computation or a similar algorithm.

In this paper we propose two new validation methods that considerably reduce the training time of the SVMr, maintaining in most cases its performance in terms of accuracy. The first new validation technique, called *percentage cross validation*, is based on splitting the initial training set into two subsets with different percentage of samples, one with *N*% of the samples and another with the rest of the samples (100 − *N*%), obtaining two models from these subsets and testing them in the complementary set. This process is repeated several times by increasing the value of *N* in each step. The second method is called *generalized predictive cross validation*, and it is based on testing the behavior of sub-models which are created using predictions of an input set. We will test the proposed cross validation methods in several regression problems, extracted from UCI and StatLib machine learning repositories. Moreover, we will solve all the problems considered with both the standard SVMr and the multi-parametric SVMr with evolutionary training in order to show the goodness of the proposed cross validation methods in both cases.

The structure of the rest of the paper is the following: next section presents the mathematical foundations of the standard SVMr and the multi-parametric model considered in this paper. Section 3 shows the validations method proposed as an alternative to traditional methods. *K*-fold cross validation is also introduced in this section as a reference (standard) algorithm for cross validation. Section 4 presents the performance of the two different SVMr approaches with every validation method in several real regression problems. Finally, Section 5 closes the paper giving some remarks.

2. Standard ϵ-SVM formulation and the multi-parametric approach

The ϵ-SVM method for the SVMr (Smola & Schölkopf, 1998) consists of, given a set of training vectors $S = \{(\mathbf{x_i}, y_i), i = 1,\ldots,l\}$, obtaining a model of the form $y(\mathbf{x}) = f(\mathbf{x}) + b = \mathbf{w}^T\phi(\mathbf{x}) + b$, to minimize a general risk function of the form

$$R[f] = \frac{1}{2}\|\mathbf{w}\|^2 + \frac{1}{2}C\sum_{i=1}^{l} L(y_i, f(\mathbf{x})), \tag{1}$$

where $\mathbf{w}$ controls the smoothness of the model, $\phi(\mathbf{x})$ is a function of projection of the input space to the feature space, b is a parameter of bias, $\mathbf{x_i}$ is a feature vector of the input space with dimension N, y_i is the output value to be estimated and $L(y_i,f(\mathbf{x}))$ is the selected loss function. In this paper, we use the L1-SVR (L1 support vector regression), characterized by an ϵ-insensitive loss function (Smola & Schölkopf, 1998)

$$L(y_i, f(\mathbf{x})) = |y_i - f(\mathbf{x_i})|_\epsilon. \tag{2}$$

In order to train this model, it is necessary to solve the following optimization problem (Smola & Schölkopf, 1998):

$$\min \quad \left(\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*)\right) \tag{3}$$

$$\text{subject to} \quad y_i - \mathbf{w}^T\phi(\mathbf{x_i}) - b \leqslant \epsilon + \xi_i, \quad i = 1,\ldots,l, \tag{4}$$

$$-y_i + \mathbf{w}^T\phi(\mathbf{x_i}) + b \leqslant \epsilon + \xi_i^*, \quad i = 1,\ldots,l, \tag{5}$$

$$\xi_i, \xi_i^* \geqslant 0, \quad i = 1,\ldots,l. \tag{6}$$

The dual form of this optimization problem is usually obtained through the minimization of the Lagrange function, constructed from the objective function and the problem constraints. In this case, the dual form of the optimization problem is the following:

$$\max \quad \left(-\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)\left(\alpha_j - \alpha_j^*\right)K(\mathbf{x_i},\mathbf{x_j}) - \epsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*)\right) \tag{7}$$

$$\text{subject to} \quad \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0, \tag{8}$$

$$\alpha_i, \alpha_i^* \in [0, C]. \tag{9}$$

In addition to these constraints, the Karush–Kuhn–Tucker conditions must be fulfilled, and also the bias variable, b, must be obtained. We do not detail this process for simplicity, the interested reader can consult Smola and Schölkopf (1998) for reference. In the dual formulation of the problem the function $K(\mathbf{x_i},\mathbf{x_j})$ is the kernel matrix, which is formed by the evaluation of a kernel function, equivalent to the dot product $\langle\phi(\mathbf{x_i}),\phi(\mathbf{x_j})\rangle$. An usual election for this kernel function is a Gaussian function, as follows:

$$K(\mathbf{x_i},\mathbf{x_j}) = exp(-\gamma\cdot\|\mathbf{x_i} - \mathbf{x_j}\|^2). \tag{10}$$

The final form of function $f(\mathbf{x})$ depends on the Lagrange multipliers α_i, α_i^*, as follows:

$$f(\mathbf{x}) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)K(\mathbf{x_i},\mathbf{x}). \tag{11}$$

2.1. Extension to multi-parametric Gaussian kernels

Several authors have considered a variation of the Support Vector Machine algorithm by including a multi-parametric Gaussian kernel function, with different values of the γ parameter in each direction in the feature space (Friedrichs & Igel, 2005; Rojas & Fernández-Reyes, 2005). This Gaussian kernel function is given by the following expression:

$$\begin{aligned} K(x_i,x_j) &= exp(-(x_i - x_j)^T Q(x_i - x_j)) \\ &= exp\left(-\sum_{m=1}^{N}\sum_{n=1}^{N}\gamma_{mn}(x_{im} - x_{jm})(x_{in} - x_{in})\right), \end{aligned} \tag{12}$$

where N is the number of dimensions in the feature space, Q is the matrix with size N that represents the width of the Gaussian function in each direction in the feature space and γ_{mn} is the element of the matrix Q which associates feature m with feature n. Without loss of generality, in this work we consider that matrix Q is a diagonal matrix with different values in each position of the main diagonal, so the simplified Gaussian kernel considered is then:

$$K(x_i,x_j) = exp\left(-\sum_{m=1}^{N}\gamma_m(x_{im} - x_{jm})^2\right). \tag{13}$$

Thus, for the case of regression problems, the hyper-parameters of the multi-parametric kernel SVMr are C, ϵ and γ_m, $m = 1,\ldots,N$.