Expert Systems with Applications 37 (2010) 5577–5589

Contents lists available at ScienceDirect

Expert Systems with Applications

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

Support vector regression based hybrid rule extraction methods for forecasting

M.A.H. Farquad [a,b], V. Ravi [a,*], S. Bapi Raju [b]

[a] Institute for Development and Research in Banking Technology, Castle Hills Road #1, Masab Tank, Hyderabad 500 057 (AP), India
[b] Department of Computer and Information Sciences, University of Hyderabad, Hyderabad 500 046 (AP), India

A R T I C L E I N F O

Keywords:
Rule extraction
Support vector machine
Support vector regression
Classification and regression tree
Adaptive network based fuzzy inference system
Dynamic evolving fuzzy inference system
Root mean squared error

A B S T R A C T

Support Vector Regression (SVR) solves regression problems based on the concept of Support Vector Machine (SVM) introduced by Vapnik (1995). The main drawback of these newer techniques is their lack of interpretability. In other words, it is difficult for the human analyst to understand the knowledge learnt by these models during training. The most popular way to overcome this difficulty is to extract *if–then* rules from SVM and SVR. Rules provide explanation capability to these models and improve the comprehensibility of the system. Over the last decade, different algorithms for extracting rules from SVM have been developed. However rule extraction from SVR is not widely available yet. In this paper a novel hybrid approach for extracting rules from SVR is presented. The proposed hybrid rule extraction procedure has two phases: (1) Obtain the reduced training set in the form of support vectors using SVR (2) Train the machine leaning techniques (with explanation capability) using the reduced training set. Machine learning techniques viz., Classification And Regression Tree (CART), Adaptive Network based Fuzzy Inference System (ANFIS) and Dynamic Evolving Fuzzy Inference System (DENFIS) are used in the phase 2. The proposed hybrid rule extraction procedure is compared to stand-alone CART, ANFIS and DENFIS. Extensive experiments are conducted on five benchmark data sets viz. Auto MPG, Body Fat, Boston Housing, Forest Fires and Pollution, to demonstrate the effectiveness of the proposed approach in generating accurate regression rules. The efficiency of these techniques is measured using Root Mean Squared Error (RMSE). From the results obtained, it is concluded that when the support vectors with the corresponding predicted target values are used, the SVR based hybrids outperform the stand-alone intelligent techniques and also the case when the support vectors with the corresponding actual target values are used.

1. Introduction

During the last decade a number of researchers and practitioners are using SVM to solve pattern classification and function-approximation problems. Function approximation or regression problems, unlike classification problems, have continuous output variable. In many applications such as medicine, finance it is desirable to extract knowledge from SVM for the users to gain a better understanding of how the model solves the problems. SVM algorithm developed by Vapnik (1995) is based on statistical learning theory. For classification problems (Burges, 1998; Osuna, Freund, & Girosi, 1997; Pontil & Verri, 1997) it tries to find a maximal margin hyperplane that separates two classes. In the case of regression problem the goal is to construct a hyperplane that lies near to or close to as many instances as possible (Ancona, 1999; Joachims, 1998; Smola & Scholkopf, 1998). Therefore the objective is to find the hyperplane with small norm while simultaneously minimizing the sum of the distances from data points to the hyperplane. Despite their superior performance in various application areas, the models created by SVM and SVR are opaque, which must be considered as a serious drawback as these models do not yield human comprehensible knowledge. Many researchers have tried to treat this accuracy vs. comprehensibility trade-off by converting the opaque, high accurate model to transparent model via rule extraction.

Rule extraction algorithms were first developed in the context of neural networks. They provide transparency for the opaque models (Gallant, 1988). Researchers argued that even limited explanation can positively influence the system acceptance by the user (Davis, Buchanan, & Shortliffe, 1977). Using rule extraction a learning system might discover salient features in the input data whose importance was not previously recognized (Craven & Shavlik, 1994).

Multiple rule extraction techniques have been proposed by the researchers to extract rules from SVM. Nunez, Angulo, and Catata (2002) proposed SVM + Prototype for extracting rules from SVM.

* Corresponding author. Tel.: +91 40 23534981x2042.
E-mail addresses: farquadonline@gmail.com (M.A.H. Farquad), rav_padma@yahoo.com (V. Ravi), bapics@uohyd.ernet.in (S.B. Raju).

0957-4174/$ - see front matter
doi:10.1016/j.eswa.2010.02.055

They used K-means clustering algorithm for determining prototype vectors for each input class. An ellipsoid is defined in the input space combining these prototypes with support vectors and mapped it to *if–then* rules. Fung, Sandilya, and Bharat Rao (2005) proposed a rule extraction technique similar to SVM + Prototype. But it does not require the computationally expensive clustering. Instead, the algorithm transforms the problem to a simpler, equivalent variant and constructs the hyper cubes by solving linear programs. Each hypercube is then transformed to a rule.

Fu, Ong, Keerthi, Hung, and Goh (2004) proposed RulExtSVM for extracting *if–then* rules using intervals defined by hyper rectangular forms. A hyper rectangle is generated by using the intersection of the support vectors with the decision boundary and the rules are generated based on the hyper rectangles. Initial rule set1 is then tuned in order to improve rule accuracy and the redundant rules are removed to obtain more concise rule set. The disadvantage of this algorithm is the construction of hyper rectangles based on the number of support vectors. A hybrid rule extraction technique is proposed by Barakat and Diederich (2004, 2005), where they first developed an SVM model using the training set and removed the output class labels. They used the developed model to predict the output class labels. Later, they used these support vectors for training the decision tree and generated rules. Hyper rectangle Rules Extraction (HRE) (Zhang, Su, Jia, & Chu, 2005) first constructs hyper rectangles according to the prototypes and the support vectors (SVs) using SVM model. By projecting these hyper rectangles onto coordinate axes, *if–then* rules are obtained. Barakat and Bradley (2006) describe the use of the area under the receiver operating characteristic (ROC) curve (AUC) to assess the quality of rules extracted from SVM.

Fuzzy Rule Extraction (FREx) (Chaves, Vellasco, & Tanscheit, 2005) determines the projection of the support vectors in the coordinate axes during first step. Using triangular fuzzy membership function each support vector is then transformed into fuzzy *if–then* rule. A Multiple Kernel-Support Vector Machine (MK-SVM) scheme with feature selection, rule extraction and prediction modeling is proposed to improve the explanation capacity of SVM (Chen, Li, & Wei, 2007). This approach makes use of the information provided by the separating hyperplane and support vectors. Rules obtained from the present approach have good generalization capacity and comprehensibility and are applied to gene expression data of cancer tissue. Barakat and Bradley (2007) proposed a method to extract rules directly from the support vectors (SVs) of a trained SVM using a modified sequential covering algorithm termed SQRex–SVM. Rules are generated based on an ordered search of the most discriminative features, as measured by interclass separation. Rule set performance is then evaluated using the measured rates of true positives (TPs) and false positives (FPs), and the AUC. Martens, Baesens, and Gestel (2009) proposed a new Active Learning-Based Approach (ALBA) to extract rules from SVM models. ALBA makes use of the key concept of SVM that the support vectors are typically close to decision boundary and extracts rules from the trained SVM. All these methods except for Barakat and Diederich (2004, 2005), do not employ hybrid approach for rule extraction. Barakat and Diederich (2004, 2005) applied decision trees to extract rules from a trained SVM. However, they did not test their approach sufficiently.

Then Farquad, Ravi, and Raju (2008a, 2008b) proposed a hybrid rule extraction approach using SVM for bankruptcy prediction in banks. They used SVM as a preprocessor to extract support vectors. These support vectors along with their corresponding actual output values of the target variable are used to train a classifier with explanation capability such as Fuzzy Rule Based System (FRBS), Decision Tree and RBF in the second phase. It is observed from the empirical results that the hybrid SVM + FRBS outperformed the stand-alone classifiers.

In this paper, we present a hybrid rule extraction procedure for solving regression problems. The proposed approach has two phases, (i) SVR is used to extract the support vectors from the training set. Two different training sets are made, where, one set has support vectors along with their corresponding actual output values given in the dataset and the second set has the support vectors and their corresponding output values predicted by SVR. (ii) These two sets are then used separately to generate rules using CART, ANFIS and DENFIS. By using the first training set, we are reducing the number of patterns in the input space (because we used support vectors only) and the rules are generated in phase 2 are not extracted from SVR. However by using the 2nd training set, we are ensuring that the rules generated in phase 2 are indeed extracted from SVR.

The rest of the paper is organized as follows. Section 2 presents overview of SVR, CART, ANFIS and DENFIS. Section 3 explains the architecture of proposed hybrid approach. Section 4 presents the experimental setup. The datasets used in the study are described in Section 5. Section 6 presents results and discussion. Finally Section 7 concludes the paper.

2. Overview of the intelligent and machine learning techniques

2.1. Support vector regression

The support vector machine (SVM) is a constructive learning procedure based on the statistical learning theory (Vapnik, 1995). SVMs are an inductive machine learning technique based on the structural risk minimization principle that aims at minimizing the true error. An SVM performs classification by constructing an N-dimensional hyper plane that optimally separates the data into two categories. The main objective of SVM is to find an optimal separating hyperplane that correctly classifies data points as much as possible and separates the points of two classes as far as possible, by minimizing the risk of misclassifying the training samples and unseen test samples.

In a typical regression problem, we are given a training set $\{(x_i, y_i)\}_{i=1}^{n} \subset \mathbb{R}^d \times \mathbb{R}$, where x_i and y_i are the input variable vector and output variable, respectively, of the ith pair. Support vector regression (Scholkopf & Smola, 2002) is a kernel method that performs nonlinear regression based on the kernel trick. Essentially, each input $x_i \in \mathbb{R}^d$ is mapped implicitly via a nonlinear feature map $\phi(\cdot)$ to some kernel-induced feature space F where linear regression is performed.

In SVR (Smola & Schölkopf, 2004; Vapnik, 1995), the goal is to find a function $f(x)$ that has at most ε deviation from the actually obtained targets y_i for all the training data. Deviation larger than ε is not accepted.

In the case of linear functions f taking the form

$$f(x) = \sum_{i=1}^{n} w_i x_i + b \quad \text{with } w \in \mathbb{R}, b \in \mathbb{R}. \tag{1}$$

One way to ensure minimum ε deviation is to minimize the norm,

$$\text{i.e.} \quad \|w\|^2 = \sum_{i=1}^{n} w_i^T w_i.$$

The problem can be written as a convex optimization problem:

$$\begin{aligned} &\text{minimize} && \frac{1}{2}\|w\|^2 \\ &\text{subject to} && \begin{cases} y_i - \sum_{i=1}^{n} w_i x_i - b \leqslant \varepsilon, \\ \sum_{i=1}^{n} w_i x_i + b - y_i \leqslant \varepsilon. \end{cases} \end{aligned} \tag{2}$$