Chemometrics and Intelligent Laboratory Systems 120 (2013) 71–76

Contents lists available at SciVerse ScienceDirect

Chemometrics and Intelligent Laboratory Systems

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

A novel tree kernel support vector machine classifier for modeling the relationship between bioactivity and molecular descriptors

Xin Huang [a,b,1], Dong-Sheng Cao [c,1], Qing-Song Xu [a,*], Liang Shen [a], Jian-Hua Huang [c], Yi-Zeng Liang [c]

[a] School of Mathematics and Statistics, Central South University, Changsha 410075, PR China
[b] Department of Mathematics, Hunan City University, Yiyang, 413000, PR China
[c] Research Center of Modernization of Traditional Chinese Medicine, Central South University, Changsha 410083, PR China

ARTICLE INFO

Article history:
Received 6 September 2012
Accepted 11 November 2012
Available online 23 November 2012

Keywords:
Tree kernel
Support vector machine (SVM)
Structure–activity relationship (SAR)
Monte Carlo
Classification and regression tree (CART)

ABSTRACT

Support vector machine (SVM) has been gaining popularity in the field of chemistry. However, it also suffered from the problems of feature subset selection in most of applications. In the present study, we attempt to construct an informative novel tree kernel to address these problems. The constructed tree kernel can effectively discover the similarities of samples and handle nonlinear classification problems. Simultaneously, informative features can be evaluated by variable importance ranking in the process of building kernel by a large number of decision trees. Thus, under the framework of kernel methods, a novel tree kernel support vector machine (TKSVM) has been proposed to model the structure–activity relationship between bioactivities and molecular structures. Three datasets related to different categorical bioactivities of compounds are used to test the performance of TKSVM. The results show that the present method is a promising one compared to the SVM models with other commonly used kernels.

1. Introduction

Structure–activity relationship (SAR), as a very important area in the modern pharmaceutical industry, is urgently needed for predicting ADME/T (absorption, distribution, metabolism, excretion and toxicity) properties to select lead compounds for optimization at the early stage of drug discovery and to screen drug candidates for clinical trials [1]. Much effort in recent SAR studies has been focused on predicting pharmacokinetic and toxicological properties that are collectively referred to as ADME/T of compounds [2–6]. The aim of the SAR analysis is to investigate and construct the relationship between chemical structure and biological activity. Up to date, many SAR modeling approaches have been reported to describe and construct the relationship, including multivariate linear regression (MLR), principal component regression (PCR) [7,8], partial least squares discriminant analysis (PLSDA) [9], *k*-nearest neighbor (k-NN) [10], classification and regression tree (CART) [11], and recently support vector machine (SVM) [12–16]. Among all these modeling methods, SVM is gaining popularity in a wide variety of the SAR study due to its prediction performance. However, many researchers have pointed out that SVM also suffered from the problem of feature subset selection [17–19]. Typically, redundant descriptors may destroy the patterns contained in the SAR data and subsequently achieve a poor prediction model. The current solution is the application to feature selection techniques to filter some uninformative or unrelated features before SVM is performed, such as filter methods and wrapper methods. How to effectively extract the patterns and improve the prediction ability for SVM appears to be still very necessary in the SAR study.

Kernel methods have been the effective tool to solve nonlinear problems in chemistry. A representative example is SVM. In our previous study, we have pointed out the modularity of kernel methods. That is, representation of data can be separated from the design of modeling algorithms. The data being considered can be represented by using a suitable kernel. Thus, we can independently build a more flexible and powerful modeling algorithm only by using a specific kernel as an input. A more detailed description of this idea can be found in [20–26]. It is known that choosing a suitable kernel is of prime importance to present the data. So how to select the best kernel among this extensive of possibilities, including graph kernel, string kernel, spectrum kernel etc., becomes the most critical stage in applying kernel-based algorithms (e.g., SVMs) in practice. Likewise, whether we could construct a specific kernel to overcome the influence of uninformative variables will be the main focus of our paper.

In this work, a novel tree kernel is proposed to deal with this issue, which is based on random selection of a part of samples via the Monte Carlo procedure followed by a CART algorithm. The constructed tree kernel using CART ensemble can give an intrinsic measure of similarities between samples and effectively cope with nonlinear classification problems. Simultaneously, the informative descriptors can be successfully discovered by means of variable importance. Thus, our tree kernel makes full use of information about important variables

* Corresponding author.
E-mail address: qsxu@csu.edu.cn (Q.-S. Xu).
[1] These authors contributed equally to this paper.

0169-7439/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.chemolab.2012.11.008

and neglect the effect of those noisy variables. Under the framework of kernel methods, the tree kernel support vector machine classification algorithm (TKSVM) is explored for predicting the ADME/T properties of drug compounds. Three datasets related to different categorical bioactivities of compounds are used for evaluating the performance of TKSVM. The results obtained show that the TKSVM approach is really an attractive alternative technique in the SAR data analysis.

2. Support vector machine (SVM)

Support vector machine is originally developed by Vapnik et al. [26]. A detailed description of the theory of SVM can be easily found in several excellent books and literature [26–29]. Assuming that the dataset contains n observations $\mathbf{x_1}$, $\mathbf{x_2}$, …, $\mathbf{x_n}$, where $\mathbf{x_i}(\mathbf{i}=1, 2, \ldots, \mathbf{n})$ is a p-dimensional column vector (p is the number of the predictors of the dataset). Let $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2, \ldots,\mathbf{x_n}]^t$ be the predictor matrix and $\mathbf{y}=[y_1, y_2, \cdots y_n]^t$ be the response. For linearly separable cases, the decision function of SVM can be expressed in the following way:

$$f(\mathbf{x}_i) = \mathrm{sgn}\left(\mathbf{w}^t\mathbf{x}_i + b\right) \tag{1}$$

where $\mathbf{w}$ is a vector of weights, and b is the constant coefficient. In the original feature space, the constraint for perfect classification can be described as:

$$y_i\left(\mathbf{w}^t\mathbf{x}_i + b\right) \geq 1, i = 1, 2, \cdots, n. \tag{2}$$

The aim of SVM is to find a vector $\mathbf{w}$ and a parameter b which can be estimated by minimizing $||\mathbf{w}||^2$. This can be solved by the quadratic optimization method. Thus, the decision function of SVM can finally be written as:

$$\begin{aligned} f(\mathbf{x}) &= \mathrm{sgn}\left(\mathbf{w}^t\mathbf{x} + b\right) \\ &= \mathrm{sgn}\left(\sum_{i=1}^{n}\mathbf{y}_i\boldsymbol{\alpha}_i < \mathbf{x}_i, \mathbf{x} > + b\right) = \mathrm{sgn}\left(\boldsymbol{\alpha}^t\mathbf{M}\mathbf{K}_t + b\right) \end{aligned} \tag{3}$$

where $<\cdot>$ denotes the inner product. M is the $n\times n$ diagonal matrix with $\mathbf{M}_{ii}=y_i(i=1, 2, \cdots, n)$, $\mathbf{K}_t=(<\mathbf{x}_1, \mathbf{x}>, <\mathbf{x}_2, \mathbf{x}>, \cdots, <\mathbf{x}_n, \mathbf{x}>)^t$, $\boldsymbol{\alpha}=(\alpha_1, \alpha_2, \cdots \alpha_n)^t$ is the optimized Lagrange multiplier vector. And b can be computed using the following equation:

$$b = \mathrm{y}_j - \sum_{i=1}^{n}\mathbf{y}_i\boldsymbol{\alpha}_i < \mathbf{x}_i, \mathbf{x}_j > = \mathrm{y}_j - \boldsymbol{\alpha}^t\mathbf{M}\mathbf{K}_j, \quad j \in \left\{0 < \boldsymbol{\alpha}_j \leq C\right\}. \tag{4}$$

where $\mathbf{K}=\mathbf{X}\mathbf{X}^t$ is a linear kernel matrix, $\mathbf{K}_j$ is the j column of kernel matrix.

So far, we have obtained the general solution form of SVM based on linear kernel. In fact, we can consider modeling methods separately from the choice of kernel functions due to the modularity of kernel methods. Thus, in order to meet specific scientific tasks, we can establish different kernel SVM by using any suitable kernel instead of the linear kernel matrix $\mathbf{K}=\mathbf{X}\mathbf{X}^t$. In the following sections, we will attempt to construct a novel tree kernel to solve the problem of feature selection to a certain degree.

3. Building tree kernel using decision tree ensemble

Decision tree, or classification and regression tree (CART), proposed by Breiman et al. [11], is a nonparametric statistical technique, which has been widely used in data mining. For classification, the aim of a CART is to divide the total data space into some high class-purity segments by selecting some useful variables from variable space. CART can easily select informative descriptors from variable space and recursively divide the given total sample space into several rectangular areas with specific similarity (e.g., the samples under the same terminal node). Inspired by these advantages, with the help of CART, we constructed an informative kernel matrix (see Fig. 1), which is based on the random selection of small sample populations via Monte-Carlo selection followed by the application of the decision tree algorithm [30,31].

Suppose that we are given a kernel matrix $\mathbf{K}$ of size $n\times n$ with all elements equal to 0. To begin with, in order to guarantee that the samples for each class can be evenly divided into two parts, we randomly divided the training set samples for each class into two parts in accordance with the same size. One is the training set and the other is validation set. The training and validation sets for each class are combined to obtain the final training and validation set. Generally speaking, the size of the training set varies from 40% to 80% of the training set.

In the second step, we use the training samples to grow a classification tree and the validation samples to prune the overgrown classification tree for obtaining the optimal pruning level (e.g., L_{best}). We construct a suboptimal tree using a fuzzy pruning strategy (randomly generate a pruning level L between 1 and L_{best}), which is between optimal tree and overgrown tree, and then prune the overgrown tree by L. The fuzzy pruning strategy helps in effectively exploiting the information of internal nodes, but does not totally destroy the structure of the tree.

In the third step, all the samples are predicted by the suboptimal tree, and thereby each sample falls into one of the terminal nodes. It is worth noting that the samples under the same terminal node may have some specific similarity to some extent besides class similarity. If two samples i and j fall into in the same terminal node, the sample similarity measure $\mathbf{K}(i,j)$ is increased by one. Thus, $\mathbf{K}$ can be considered as a similarity measure and reflects the size of similarity among these training samples. After that, we can repeat the above process many times (e.g., *ntree*) so that a lot of tree models are established. Accordingly, the kernel matrix is changed by the results of the tree. At the end, the kernel matrix is normalized by dividing by the number of trees (ntree). Note that the similarity between a sample and itself is always set to one (i.e., $\mathbf{K}(i,i)=1$).

Likewise, we can construct the predictive kernel matrix between training samples and m new test samples as follows: A test kernel matrix $\mathbf{K}_t$ of size $n\times m$ with all elements equal to 0 is firstly generated. Then these m new test samples are predicted by the established trees, and each new test sample turns up one of the terminal nodes in each tree. Likewise, if the new test sample j and some training sample i turn up in the same terminal node, the predictive kernel matrix $\mathbf{K}_t(i,j)$ is increased by one. $\mathbf{K}_t(i,j)$ reflects the degree of similarity between new sample j and training sample i. The bigger the $\mathbf{K}_t(i,j)$ is, the more similar they are.

4. Evaluation of variable importance

The importance of a variable in CART can be determined by the decrease of impurity across the tree for all non-terminal nodes that use this variable as a splitter [11]. Based on the kernel constructed above, tree kernel mainly incorporates the idea of ensemble variable selection. The variable importance in TKSVM can be obtained by averaging variable importance of all decision trees, which is given as follows:

$$VIP(j) = \frac{1}{ntree}\sum_{i=1}^{ntree} VIP_{t_i}(j) \tag{5}$$

where *ntree* is the number of trees in the process of building kernel, t_i denotes the ith decision tree, and $VIP_{t_i}(j)$ is the importance of the jth variable in the ith decision tree.