Automatic feature subset selection for decision tree-based ensemble methods in the prediction of bioactivity

Dong-Sheng Cao a, Qing-Song Xu b, Yi-Zeng Liang a,*, Xian Chen a, Hong-Dong Li a

A R T I C L E   I N F O

Article history:
Received 8 January 2010
Received in revised form 15 June 2010
Accepted 23 June 2010
Available online 7 July 2010

Keywords:
Feature selection
Bagging
Boosting
Random Forest (RF)
Classification and Regression Tree (CART)
Ensemble learning

A B S T R A C T

In the structure–activity relationship (SAR) study, a learning algorithm is usually faced with the problem of selecting a compact subset of descriptors related to the property of interest, while ignoring the rest. This paper presents a new method of molecular descriptor selection utilizing three commonly used decision tree (DT)-based ensemble methods coupled with a backward elimination strategy (BES). Our proposed method eliminates descriptor redundancy automatically and searches for more compact descriptor subset tailored to DT-based ensemble methods. Six real SAR datasets related to different categorical bioactivities of compounds are used to evaluate the proposed method. The results obtained in this study indicate that DT-based ensemble methods coupled with BES, especially boosting tree model, yield better classification performance for compounds related to ADMET.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

In modern pharmaceutical industry, structure–activity relationship (SAR), an important area of chemometrics, is urgently needed for predicting ADMET (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 [1–8]. The aim of SAR is to search information relating the molecular structure to biological activity. In this process, molecular structures are usually represented by a variety of molecular descriptors which are easily calculated by some software packages. In most cases, the number of compounds with the biological activity values available is usually small compared with the number of descriptors. This may lead to several difficult problems for constructing a SAR model undoubtedly. Among these problems we could list: (1) In most cases only a small number of descriptors have substantial influence on the property of interest. A lot of descriptors usually include ones unrelated to the property of interest. The use of such descriptors likely generates noise in an established model, which may affect the prediction accuracy of that model. (2) It is really difficult to determine which descriptors or combinations are responsible for the property of interest. The identification of important descriptors is of fundamental and practical interest. Research in biology and medicine may benefit from the examination of the top ranking descriptors to confirm recent discoveries in new drug research or suggest new avenues to be explored. [3] Much time may be needed to conduct a learning algorithm with a very large number of descriptors. (4) Most of modeling methods may not work well or even may be invalid when too many descriptors exist in the model. A key reason is that data overfitting usually arises when the number p of the descriptors significantly exceeds the number N of the training samples or high multicollinearity among the descriptors exists. In such a situation, one can easily find a decision function that separates the training samples well but performs poorly on the test data. To overcome these difficulties, feature subset selection methods can in general be carried out to partially reduce the redundancy in descriptors [9–17]. Feature subset selection methods are the process identifying and removing as many of irrelevant and redundant descriptors as possible. Exhaustive enumeration is also impractical for large numbers of features because of the combinatorial explosion of the number of subsets. Thus methods capable of automatic descriptor selection are of practical importance to fast select a small subset of descriptors.

Currently, several ensemble learning methods, such as bagging, boosting and stacking etc, have been successfully applied to explore the relationship between molecular structures and the properties of interest in SAR study [13,18–26]. These methods can remarkably improve the prediction performance of models by combining rough or moderately rough rules of thumb. A few well-known examples are decision tree (DT)-based ensemble methods including bagging tree, random forest and boosting tree etc. DT-based ensemble methods can overcome the major drawback (i.e. low prediction ability and
instability) of single DT, but fortunately retain most of its advantages. For example, they are capable of dealing with problems where the number of descriptors is larger than the number of compounds and/or where the large majority of descriptors are irrelevant. Moreover, they can also be easily used to sort molecular descriptors by means of the criterion of variable importance measure called sensitivity analysis. All these advantages make DT-based ensemble methods more and more popular in SAR study. However, single decision tree is also known to degrade in prediction accuracy when faced with a number of unimportant features [27]. Whether DT-based ensemble algorithms also need to select the subset of features is worth discussing and studying. So, it is of practical necessity to study the prediction performance of DT-based ensemble algorithms with and without feature selection in SAR study.

In the paper, we study the problem of selecting a compact feature subset from a large pool of molecular descriptors. We propose a new molecular descriptor selection method utilizing DT-based ensemble methods coupled with a backward elimination strategy (BES). Six real SAR datasets related to different categorical bioactivities of compounds are used to evaluate the proposed method. The results demonstrated that our proposed method can eliminate descriptor redundancy automatically and yield more compact descriptor subset tailored to DT-based ensemble methods. We also experimentally demonstrated that the molecular descriptors selected by our technique give superior prediction accuracies.

2. Methods and materials

2.1. Classification and Regression Tree

Classification and Regression Trees (CART), proposed by Breiman et al. [28], is a nonparametric statistical technique. The goal of CART is to explain the response y by selecting some useful variables from a large pool of variables. The response y can be either numerical or categorical, respectively, resulting in regression or classification tree. Here, only a concise description of classification tree is presented. The CART procedure is generally made up of three steps. In the first step, the full tree is built using a binary split procedure. Starting from the root node including all the training samples, a yes-no question at some variable x_{i} is asked, and the samples for which the answer is “yes” (x_{i} ≤ θ, θ is a cut-off value for this given variable) are assigned to the left branch and the others (x_{i} > θ) are assigned to the right branch. Thus, a parent node is divided into two child nodes, each of which can then be subdivided independently. We can repeat the above process certain times until some stopping criterion (e.g., each node includes at least a predetermined number of samples or contains only objects of one class) is reached. Thus each sample is assigned to one of the terminal nodes (the terminal nodes refer to nodes having no child nodes) based on the answers to the questions. The full tree may be an overgrown model, which closely describes the training set and generally shows overfitting (i.e., the training samples are split well but the prediction for unknown samples is poor). A solution to this problem is to prune the full tree. In the second step, the overfitted tree is pruned. For a given tree, the total number L of terminal nodes represents the size of a tree. For a given number of terminal nodes, pruning could lead to several different sub-trees with a smaller number of terminal nodes. Pruning starts from the bottom of a tree. In each pruning step a pair of terminal nodes from a parent node is pruned away. We repeat the above pruning step several times so that a set of sub-trees with different terminal nodes are reached. The third step of CART is to select an optimal sub-tree based on the quality of prediction for new samples. This is done according to a cost complexity criterion.

\[ C_{γ}(M) = Q(M) + γL \]  

Here, Q(M) is the misclassification cost of the sub-tree obtained by pruning and the regularization parameter γ determines the trade-off between the overall misclassification cost and the model complexity measured by L. The value of γ can usually be chosen by cross validation.

2.2. Bagging tree

Breiman’s bagging [29], short for bootstrap aggregating, is one of the earliest and simplest ensemble algorithms with a surprisingly good performance. The bagging tree builds tree ensembles (CART algorithm without pruning) from bootstrap samples (i.e., samples of the same size as the training set drawn with replacement from it), and then combines all classifiers to a single one.

Given a data set on N molecules for training, \( Z = \{(x_{1}, y_{1}), (x_{2}, y_{2}), \ldots, (x_{N}, y_{N})\} \), where \( x_{i} \) is the number of molecules and \( y_{i} \) is the corresponding class label. The bagging tree can be implemented as follows:

1. Repeat for \( t = 1, 2, \ldots, T \):
   a. \( N \) samples are randomly picked from the training set with replacement and we can get a “bootstrap set” \( Z^{*} = \{(x_{1}^{*}, y_{1}), (x_{2}^{*}, y_{2}), \ldots, (x_{N}^{*}, y_{N})\} \). In this process, the overlapping of the original samples is allowed.
   b. Decision tree model (CART algorithm without pruning) is built with the “bootstrap set” selected in above step to generate predictions of \( N \) samples.

2. Ensemble prediction of each sample is obtained. For classification problems, the bagging estimate is the class predicted by the majority of several models.

Bagging seems to work especially well for unstable classifiers (e.g., DT) in that it can dramatically reduce the variance of these algorithms with the help of averaging [29].

2.3. Random forest

Random forest (RF) [30], as its name indicates, is also a tree ensemble approach. It has been widely applied to SAR study [31,32]. RF directly derives from bagging and so can be seen as a variant of bagging. The only difference between RF and bagging tree is the step 1(b) in the bagging tree procedure. For each bootstrap sample, RF grows a tree with the following modification: at each node, choose the best split among a randomly selected subset of \( m_{t r} \) (rather than all) features. Here, \( m_{t r} \) is essentially the only tuning parameter in RF. The tree is grown to the maximum size until some stopping criterion is reached and needs no pruning. So, each analogous tree (unpruned CART algorithm) in RF is constructed using the bootstrap samples of the training data and random feature selection in tree induction. After the number \( T \) of iterations is achieved, predictions are made by aggregating (majority vote) the predictions of all analogous trees. The other process is consistent with the algorithm of the bagging tree except for the modification of step 1(b).

2.4. Boosting tree

The other commonly used tree ensemble method is boosting [33–36]. Boosting is a power technique by combining multiple base classifiers to produce a form of committee whose performance can be significantly better than that of any of the base classifiers. The boosting algorithm in this paper is the most widely used form of the boosting algorithms called adaptive boosting or AdaBoost for short, developed by Freund and Schapire [33,34].

Consider a two-class classification problem, one class is denoted by \( y = 1 \) and another is denoted by \( y = -1 \). A classifier \( G(x):\{−1, 1\} \) can be obtained by applying the existing classification method (e.g.,
decision tree) to the training samples. The algorithm of AdaBoost can be executed in the following way.

1. Initiate the data weighting coefficients \( \{\omega_n\} \) by setting \( \omega_n^{(1)} = 1/N \) for \( n = 1, ..., N \).
2. For \( t = 1, ..., T \):
   (a) Fit a tree classifier \( G_t(x) \) to the training data by minimizing the weight error function.
   \[
   J_t = \sum_{n=1}^{N} \omega_n^{(t)} I(G_t(x_n) \neq y_n)
   \]
   (2)
   where \( I(G_t(x_n) \neq y_n) \) is the indicator function and equals 1 when \( G_t(x_n) \neq y_n \) and 0 otherwise.
   (b) Evaluate the quantities
   \[
   \varepsilon_t = \frac{\sum_{n=1}^{N} \omega_n^{(t)} I(G_t(x_n) \neq y_n)}{\sum_{n=1}^{N} \omega_n^{(t)}}
   \]
   (3)
   And then use these to evaluate confidence index
   \[
   \alpha_t = \ln \left( \frac{1 - \varepsilon_t}{\varepsilon_t} \right)
   \]
   (4)
   (c) Update the data weighting coefficients
   \[
   \omega_n^{(t+1)} = \omega_n^{(t)} \exp(\alpha_t I(G_t(x_n) \neq y_n))
   \]
   (5)
3. Combining all classifiers to construct the final model, which is given by
   \[
   G(x) = \left( \sum_{t=1}^{T} \alpha_t G_t(x) \right)
   \]
   (6)

In AdaBoost, each sample is assigned to a weight \( \omega_n^{(t)} \) that is changed at each iteration. Initially, the first classifier \( G_1(x) \) is trained using equal weighting coefficients \( \omega_n^{(1)} \), which therefore corresponds to the usual procedure for a single classifier. At the \( t \)th round (\( t = 1, 2, ..., T \)), the classifier \( G_t(x) \) can be constructed by applying the classification algorithm to the training samples with current weights \( \omega_n^{(t)} \) at step 2(a). The weighted error rates \( \varepsilon_t \) of each base classifier are computed by Eq. (3) at step 2(b). The confidence index \( \alpha_t \) for the \( t \)th classifier is calculated by Eq. (4). A large confidence index indicates that corresponding base classifier plays a more important role in the final decision. At step 2(c), the weights of all samples are updated according to such a rule: increasing the weights of samples that are misclassified by the current classifier \( G_t(x) \) and decreasing the weights for those correctly classified. So the next classifier \( G_{t+1}(x) \) will therefore place a greater emphasis on those samples misclassified by the previous ones in the sequence. After repeating the above steps (step 2(a-c)) \( T \) times, the final classifier \( G(x) \) can be obtained by combining all base classifiers with different weights \( \alpha_1, \alpha_2, ..., \alpha_T \).

2.5. Variable importance ranking

The commonly used DT-based algorithms allow to easily calculate a variable importance measure for a given classification problem [37]. In a tree classifier \( M \) (containing \( L \) terminal nodes), an importance measure of the candidate variable \( x_{c(j)} \) can be obtained according to the following criterion:

\[
J_M(x_{c(j)}) = \sum_{m=1}^{L} \Delta \xi_m I(v(m) = x_{c(j)})
\]

(7)

Here \( v(m) \) denotes the selected split variable at the node \( m \), if the variable \( x_{c(j)} \) is selected as splitting variable at the node \( m \), \( I(v(m) = x_{c(j)}) = 1 \), otherwise, \( I(v(m) = x_{c(j)}) = 0 \). The corresponding impurity decrease \( \Delta \xi_m \) evaluates the importance of variable selected to split the region at the node \( m \). If parent node \( m \) is split into two child nodes (\( m_l \) and \( m_r \)) at the node \( m \), the impurity decrease \( \Delta \xi_m \) at the node \( m \) is defined as:

\[
\Delta \xi_m = \xi_m - p_l \xi_{m_l} - p_r \xi_{m_r}
\]

(8)

where \( \xi_m, \xi_{m_l}, \text{and} \xi_{m_r} \) are the impurity at parent node \( m \), left child node and right child node, respectively, and \( p_l \) and \( p_r \) are the proportions of the samples which fall into the left child node and right child node, respectively.

Variable importance in single DT can be easily extended to the tree-based ensemble algorithms [8]. Since these tree-based ensemble algorithms are constructed from the combination of a set of DTs by averaging or a weighted vote, the variable importance in these models can be obtained by averaging or weighted averaging variable importance of all DTs, which is given as follows:

\[
J_M(x_{c(j)}) = \frac{1}{K} \sum_{k=1}^{K} J_M(x_{c(j)})
\]

(9)

where \( K \) is the number of trees in the tree-based ensemble, \( M_k \) denotes the \( k \)th classification tree.

2.6. Feature subset selection

Based on the variable importance, we construct a general framework based on the backward elimination strategy (BES) to select the optimal feature subset in this study. Fig. 1 shows the flow chart for BES. BES

![Fig. 1. The flowchart of feature selection using DT-based ensemble methods.](image-url)
mainly consists of three successive steps. In the first step, DT-based ensemble algorithms are employed to generate a model, and accordingly the corresponding variable importance and fitness are recorded. The fitness function is defined by:

\[
\text{Fitness}_i = \frac{1}{10} \sum_{k=1}^{10} \text{error}_k + \lambda |p_i|
\]

(10)

where \(|p_i|\) denotes the number of molecular descriptors in the \(i\)th iteration, and \(\lambda\) defines the tradeoff between average error rate and model complexity measured by \(|p_i|\). The aim of the complexity penalty is to penalize feature subsets with many features so as to break ties in favor of smaller subsets. The value of \(\lambda\) was set to 0.1 for these datasets with small number of features, and 0.05% for these datasets with large number of features in our study. It was added to pick the smaller of two feature subsets that have the same or similar estimated accuracy. In the fitness function, the first term denotes the average cross validation (CV) error rate, which aims to stabilize the CV error rate by averaging. The difference of CV error rate mainly derives from the influence of two part of randomization. The first one is that CV randomly splits the whole dataset into different validation set in each round. The second one is that DT-based ensemble methods make use of resampling techniques to establish each tree classifier. So, to make the estimated accuracy more accurate and make the fitness function much smoother, we average ten CV error rates as the first term of fitness through repeating ten cross-validation procedures.

In the second step, an exponentially decreasing function (EDF) is utilized to remove the descriptors with relatively small variable importance by force. In the \(i\)th iteration, the ratio of variables to be kept can be determined using an EDF defined as:

\[
r_i = a \times \exp(-k \times i)
\]

(11)

where \(a\) and \(k\) are constants determined by the following two conditions: (1) if all descriptors are included in the model firstly, \(r_i\) is set to 1. (2) In the \(B\)th iteration, only 5% descriptors are reserved such that we have \(r_B = 5\%\). With the two conditions, \(a\) and \(k\) can be calculated as:

\[
a = (r_B)^{-1/(B-1)}
\]

(12)

\[
k = -\ln(r_B) / (B-1)
\]

(13)

where \(\ln(\cdot)\) denotes the logarithmic function. Fig. 2 illustrates an example of EDF. As can be seen clearly, the process of descriptors reduction can be roughly divided into two stages. In the first stage, molecular descriptors are eliminated rapidly which performs a ‘fast selection’, whereas in the second stage, descriptors are reduced in a very gentle manner, which is called a ‘refined selection’ stage in our study. Therefore, descriptors of little or no information can be removed in a stepwise and efficient way because of the advantage of EDF. The above two steps are repeated \(B\) times so that the certain number of features to be kept is achieved (e.g., \(r_B = 5\%\)). In the third step of BES, after \(B\) iteration, \(B\) subsets of descriptors and \(B\) fitness values are obtained. The fitness estimates (by cross validation) of the resulting sequence of models are calculated so as to determine a learning curve which, typically, first decreases then reach a minimum and increases again. So, we can select the subset with the minimum fitness as the optimal subset of descriptors. A significant advantage for BES is that it is possible to use the ranking to define nested subsets of features \(F_1 \subset F_2 \subset \ldots \subset F\) and select an optimum subset of features with a model selection criterion (Eq. (10)) by varying a single parameter: the number of features.

Generally speaking, feature selection algorithms usually need three key ingredients: a mathematical modeling procedure, a search strategy, and an objective function guiding the search. In our study, DT-based ensemble methods are employed to generate a mathematical model. The search algorithm is a backward elimination approach, which refers to a search that begins at the full set of features. The main reason for this choice is that going backward from the full set of features may easily capture interacting features. Moreover, EDF can overcome the drawback of the low efficiency of BES. Nay, EDF allows us to remove chunks of features including a lot of uninformative features in the first few iterations and then remove two or three up to one feature at a time once the feature set achieves a few ones. A good search algorithm is certainly important for locating good feature subset, but the objective function is absolutely critical to obtain solutions with the desired properties. The objective function in the study is the sum of two terms: the average values of cross validation error rates and the penalty term, which play a crucial role for finding a compact subset of features.

3. Dataset

As good pharmacokinetic properties are very important for the drug candidates, there have been increasing efforts in SAR research to address the prediction of the pharmacokinetic properties of compounds, including ADMET. The studies of HIA [38,39], P-gp [40,41], TdP [42], MDDR [18,43], and BBB [44,45] are examples focusing on predicting the ADMET and adverse drug effects. We selected five datasets related to these pharmacokinetic and pharmacodynamic properties for evaluating the performance of our proposed methods. In addition, we also studied a set of compounds with their inhibition activity of Factor Xa [46,47], which is related to the blood coagulation. A brief description of the six datasets including the number of compounds and their distribution into the active and inactive classes as well as the molecule descriptors used is given in Table 1. Details about six datasets are given in the supporting information.

4. Results and discussions

4.1. Comparison of prediction accuracies using DT-based techniques with all molecular descriptors

In the study, we use 5-fold cross validation to estimate the accuracy of our models. For 5-fold cross validation, the training set is split into 5 roughly equal-sized parts firstly, and then we fit the model to four parts of the data and calculate the error rates of the other part. The process is repeated 5 times so that every part can be predicted as a validation set. For CART, we use a deviance criterion to determine the split of a tree. For bagging and RF, we use ensembles of 500 trees to obtain the better prediction performance. In addition, \(m_{tr}^\text{avg}\) is a very
important parameter affecting the prediction performance of RF. We use a default value, proposed by Breiman, which equals to the square root of the number of features. For the boosting tree, the optimal number of trees is determined by means of 5-fold cross validation. To make the prediction results more reliable, our reported accuracies are mean value of ten accuracies from 5-fold cross validation. We also show the standard deviation of the mean values as a stability measurement.

Table 2 and Fig. 3 show the comparison results of the prediction accuracies on six SAR datasets from different DT-based ensemble methods together with CART with all descriptors. As shown in Table 2 and Fig. 3, CART uniformly achieves the worst performance on six SAR datasets, which gives mean value 77.07% of prediction accuracy. This indicates that CART has poor prediction ability. Moreover, we can also see that CART has a large standard deviation on each SAR dataset, which may indicate the instability of CART. That is, a small change in the data may lead to the completely different decision boundary. However, the prediction accuracies from ensemble methods are remarkably improved by combining several decision trees. For bagging, this simplest version of ensemble methods also obtains the better performance compared with CART, which reduces the variance of the tree algorithm by combining many unpruned trees. The bagging produces the average prediction accuracy of 82.41%. For RF, this variant of bagging improves the prediction results again, which produces the average prediction accuracy of 82.93%. In the first four SAR datasets: HIA, P-gp, TdP, and MDDR including relatively small molecular descriptors, the performance by RF is superior to one by bagging. However, in BBB and Factor Xa datasets, bagging obtains somewhat better results than RF. A possible reason is that these two SAR datasets may consist of many uninformative descriptors unrelated to the properties of interest, which seriously affect the prediction ability of RF. Because RF produces a series of tree classifiers by randomly selecting certain descriptors from a large pool of descriptors, the performance of each tree classifier in RF may suffer from serious influence when RF selects few useful molecular descriptors to establish the classifier. However, from the computational point of view, RF only uses \( m_{\text{tr}} \) of the descriptors. Since \( m_{\text{tr}} \) is typically very small, the search is very fast. From Table 2 and Fig. 3, we can clearly see that for all six SAR datasets, the prediction ability of boosting is completely superior to ones from the other three methods, which indicates that boosting not only reduces the variance of single DT by averaging several decision trees obtained from different subsamples of the training samples, but also reduce the bias of single DT by forcing single DT to concentrate on hard-to-classify samples. Our results confirm earlier findings that the bagging tree almost always produces a better performance than a single decision tree and that the boosting method is a powerful technique that can usually produce better ensembles than bagging and RF. Among the three ensemble methods, the computational efficiency of RF is the fastest. In addition, it is should be noted that the standard deviation of the ten prediction accuracies obtained by three ensemble methods is small compared with one obtained by CART. It can be concluded that the classification models using ensemble methods should be more robust and reliable.

### 4.2. Effect of feature selection on overall prediction accuracies

Table 3 and Figs. 4 and 5 give the prediction accuracies of DT-based techniques for all molecular descriptors and BES-selected descriptors by 5-fold cross validation. As shown in Table 3, the prediction accuracies from DT-based ensemble methods with BES are remarkably improved compared to those from DT-based ensemble methods without BES on the average. For each SAR dataset, the results with BES are consistently better than ones without BES, which indicates that the BES indeed improves the prediction ability of DT-based ensemble methods although these DT-based ensemble methods have ability of selecting important variables. Moreover, it can be observed from Fig. 5 that boosting without BES is inferior to bagging with BES. This indicates that the ensemble methods also suffer from the problem of curse of dimensionality to some extent albeit the problem is not very serious. It should be noted that for each SAR dataset the situation is the same as the average accuracy (see Table 3). This is already very strong evidence indicating that the performance by BES is better than one by ensemble algorithms without variable selection. That is, it is of practical importance for DT-based ensemble methods to select the more useful descriptors by feature selection methods. For three DT-based ensemble methods with BES, boosting with BES achieves the best prediction ability again. The average accuracy varies from 84.85% for bagging to 86.04% for boosting. However, for bagging and RF with BES, there is no significant difference of performance. A main reason is
that the BES-selected descriptors are very useful for discrimination between two classes, and not interrelated to a large extent. RF makes use of random selection of the input variables to improve the prediction accuracy of bagging by reducing the correlation between the trees. So, when the BES-selected molecular descriptors are useful and not interrelated, RF does not exhibit more advantages than bagging and thus produces the similar results to bagging. To sum up: our results demonstrate that the BES pruning methods significantly improve the prediction performance of bagging, RF and boosting.

4.3. Comparison with other classification methods

The classification results of DT-based ensemble methods combined with BES were further evaluated by comparison with other reported methods. To make the comparison more reliable, we used the same sets of molecules and molecular descriptors as those in the referenced studies regarding the evaluation of HIA, P-gp, TdP, and MDDR. In the case of BBB and Factor Xa, we used the same whole sets of compounds as ones in the referenced studies. However, the descriptor sets used in this work were different from the referenced studies because their descriptors values were not available. Table 4 lists the results obtained from four research groups.

From Tables 3 and 4, one can see that the prediction results obtained by DT-based ensemble methods coupled with BES are better or comparable to those of SVM with recursive feature elimination (RFE) for HIA, P-gp and TdP. The average accuracies for P-gp and TdP are 82.74% and 87.65% using DT-based ensemble methods with BES, while the corresponding values computed by SVM with RFE are 79.4% and 83.9%, respectively. For dataset HIA, the average accuracy is 83.57% using boosting with BES while one of SVM with RFE is 86.7%. When using the whole molecular descriptors, DT-based ensemble methods also produced better average accuracies than SVM without RFE for datasets HIA, P-gp and TdP. For datasets P-gp and TdP, the results obtained by boosting without BES is even better than ones obtained by SVM with RFE. A main reason is that DT-based ensemble methods (i.e., RF and boosting) have the ability of selecting useful features to some extent, while SVM does not. For dataset MDDR, the classification accuracies of DT-based ensemble methods coupled with BES are better than ones obtained by various linear or nonlinear classification methods such as PLS, DT, RF and SVM etc. For datasets BBB and Factor Xa, although we used different molecular descriptors from the referenced studies, the results obtained by our proposed methods are also comparable with respect to ones from the referenced studies.

4.4. BES-selected molecular descriptors

Fig. 6 shows the plots of adjusted prediction accuracy on six SAR datasets using three different BES pruning strategies. From Fig. 6, one can see that the adjusted prediction accuracy is seen to decrease monotonically with increasing the number of BES iterations more rapidly during the early stages and then reaching a minimum and increasing again as iterations increase. The main difference of the adjusted prediction accuracy for bagging, RF and boosting occurs at the refined selection step, which indicates that the descriptors at the refined selection step are more useful and informative for discrimination. The optimal number of molecular descriptors we select is one that has the minimal adjusted prediction accuracy. Fig. 7 shows the number of selected molecular descriptors using four DT-based
algorithms coupled with BES (CART is directly computed without BES), together with the original molecular descriptors. As can be shown from Fig. 7, the selected number of molecular descriptors using DT-based ensemble methods with BES is greatly reduced. Such small descriptors can therefore significantly increase the interpretability of models. Moreover, it is somewhat surprising from Fig. 7 that the molecular descriptors selected by RF seem to be larger than ones selected by bagging. A possible reason is that by random selection of the descriptors RF can simultaneously spot such two variables, which are highly correlated; deleting one or the other of them will not affect prediction accuracy. Deleting both of them may degrade prediction accuracy considerably. But bagging does not. As an example, the BES-selected molecular descriptors by four DT-based algorithms for dataset TdP are presented in Table 5. We notice that rankings of four DT-based models are quite different, but since boosting model is much more accurate than the other ones, we deem its rankings also more reliable. We also observe that the descriptors selected by boosting are uniformly distributed in the descriptors selected by the other three models. It should be noted that two topological descriptors, in particular S(36) (Atom-type H Estate sum for $N^-$) are automatically detected by four models, which indicates that these two molecular descriptors may play a very important role for classifying the TdP compounds.

5. Conclusion

In the paper, we have proposed a systematic and flexible methodology to support the analysis and knowledge extraction from the SAR datasets. The framework can automatically be exploited to identify useful molecular descriptors related to the property of interest for drug research. From an application point of view, the

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Study</th>
<th>Methods</th>
<th>R (%)</th>
<th>SE (%)</th>
<th>SP (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIA</td>
<td>Arodz et al. [19]</td>
<td>RFSBoost-LDA$^a$</td>
<td>80.8</td>
<td>86.9</td>
<td>68.5</td>
</tr>
<tr>
<td></td>
<td>Xue et al. [42]</td>
<td>SVM$^b$</td>
<td>77.0</td>
<td>83.4</td>
<td>63.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM + RFE$^c$</td>
<td>86.7</td>
<td>90</td>
<td>80.7</td>
</tr>
<tr>
<td>P-gp</td>
<td>Arodz et al. [19]</td>
<td>RFSBoost-LDA</td>
<td>76.3</td>
<td>83.0</td>
<td>67.1</td>
</tr>
<tr>
<td></td>
<td>Xue et al. [42]</td>
<td>SVM</td>
<td>68.3</td>
<td>68.9</td>
<td>68.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM + RFE$^c$</td>
<td>79.4</td>
<td>81.2</td>
<td>79.2</td>
</tr>
<tr>
<td>TdP</td>
<td>Arodz et al. [19]</td>
<td>RFSBoost-LDA</td>
<td>74.2</td>
<td>72.9</td>
<td>74.5</td>
</tr>
<tr>
<td></td>
<td>Xue et al. [42]</td>
<td>SVM</td>
<td>82.0</td>
<td>54.5</td>
<td>90.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM + RFE$^c$</td>
<td>83.9</td>
<td>66.8</td>
<td>89.3</td>
</tr>
<tr>
<td>MDDR</td>
<td>Arodz et al. [19]</td>
<td>RFSBoost-LDA</td>
<td>81.8</td>
<td>89.2</td>
<td>72.1</td>
</tr>
<tr>
<td></td>
<td>Svetnik et al. [18]</td>
<td>SVM</td>
<td>81.5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>DT$^d$</td>
<td>77.9</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>PLS$^e$</td>
<td>81.7</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SGB$^f$</td>
<td>82.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>RF$^g$</td>
<td>83.1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM</td>
<td>79.1</td>
<td>89.9</td>
<td>64.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM + RFE$^c$</td>
<td>83.7</td>
<td>88.6</td>
<td>75.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LR$^h$</td>
<td>71.0</td>
<td>83.9</td>
<td>46.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LDA$^i$</td>
<td>71.2</td>
<td>78.2</td>
<td>58.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>C4.5 DT$^j$</td>
<td>74.3</td>
<td>80.3</td>
<td>62.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>k-NN$^k$</td>
<td>77.1</td>
<td>85.5</td>
<td>61.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PNN$^l$</td>
<td>76.5</td>
<td>84.3</td>
<td>62.1</td>
</tr>
<tr>
<td>BBB</td>
<td>Li et al. [44]</td>
<td>SVM</td>
<td>79.1</td>
<td>89.9</td>
<td>63.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SVM + RFE$^c$</td>
<td>83.7</td>
<td>88.6</td>
<td>75.0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LR$^h$</td>
<td>71.0</td>
<td>83.9</td>
<td>46.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LDA$^i$</td>
<td>71.2</td>
<td>78.2</td>
<td>58.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>C4.5 DT$^j$</td>
<td>74.3</td>
<td>80.3</td>
<td>62.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>k-NN$^k$</td>
<td>77.1</td>
<td>85.5</td>
<td>61.4</td>
</tr>
<tr>
<td>Factor Xa</td>
<td>Fontaine et al. [46]</td>
<td>PLS</td>
<td>88.0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$^a$ Random feature subset boosting for LDA.
$^b$ Support vector machine.
$^c$ SVM with recursive feature elimination.
$^d$ Decision tree.
$^e$ Partial least squares.
$^f$ Stochastic gradient boosting.
$^g$ Random forest.
$^h$ Linear regression.
$^i$ Linear discriminate analysis.
$^j$ C4.5 decision tree.
$^k$ k-nearest neighbor.
$^l$ Probabilistic neural network.

Fig. 6. Plots of adjusted prediction accuracy on six SAR datasets using three different BES pruning strategies. (A) HIA dataset (B) P-gp dataset (C) TdP dataset (D) MDDR dataset (E) BBB dataset (F) Factor Xa dataset. Solid line: Boosting + BES, Dash and dot line: Bagging + BES, Dot line: RF + BES. Along x-axis are the number of iterations in the BES procedures, and average adjusted prediction accuracy is shown along y-axis.

Fig. 7. The number of selected molecular descriptors using four DT-based algorithms coupled with BES, together with the original molecular descriptors.
framework remains as general as possible. The results obtained in this study indicate that DT-based ensemble methods coupled with BES, especially boosting tree model, give superior prediction accuracies to other reported methods for the compounds related to ADMET.

Acknowledgements
We also would like to thank the reviewers for their useful discussions, comments and suggestions throughout this entire work. This work is financially supported by the National Nature Foundation Committee of P.R. China (Grants No. 20875104 and No. 10771217), the international cooperation project on traditional Chinese medicines of ministry of science and technology of China (Grant No. 2007DFA40680). The studies meet with the approval of the university’s review board.

References