ARTICLES

A Model-Based Ensembling Approach for Developing QSARs

Qianyi Zhang,*,† Jacqueline M. Hughes-Oliver,† and Raymond T. Ng‡

Department of Statistics, North Carolina State University, Raleigh, North Carolina 27695-8203, and Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, Canada

Received February 28, 2009

Ensemble methods have become popular for QSAR modeling, but most studies have assumed balanced data, consisting of approximately equal numbers of active and inactive compounds. Cheminformatics data are often far from being balanced. We extend the application of ensemble methods to include cases of imbalance of class membership and to more adequately assess model output. Based on the extension, we propose an ensemble method called MBEnsemble that automatically determines the appropriate tuning parameters to provide reliable predictions and maximize the F-measure. Results from multiple data sets demonstrate that the proposed ensemble technique works well on imbalanced data.

INTRODUCTION

Compounds with similar chemical structure often have similar biology activity.[1] The goal of quantitative structure−activity relationship (QSAR) modeling is to determine whether chemical structures are quantitatively correlated with biology activities. A QSAR study incorporates statistical and mathematical approaches and uses computer-based tools to implement those approaches. Over the past decade, many QSAR modeling tools have been developed. Some popular examples include decision tree (DT),[2] k-nearest neighbors (KNN),[3] support vector machines (SVM),[4] neural networks (NNet),[4,5] and random forest (RF).[6,7] All of these methods are reputed for their application in QSAR modeling. However, none of them fully address all "practical" features required by QSAR modeling, including the ability to effectively handle imbalanced data and multiple mechanisms.

Typically, binary designations are used to indicate presence of the studied biological activity. Each compound is assigned a value of one or zero, with one indicating desired activity and zero indicating no or little activity. When compound collections have unequal numbers of active and inactive compounds, the resulting data on activity and structural descriptors is said to be imbalanced. In the real world, biologically active compounds tend to be rare when compared with inactive compounds, so the chemical data set is usually highly imbalanced. High-throughput screening data submitted to PubChem[8] by the Molecular Libraries Screening Centers Network[9] tend to have activity rates much less than 0.1%.

QSAR modeling is complicated by the imbalanced feature of the data. It is more difficult to predict active than to predict inactive, no matter which QSAR model is utilized. For extremely imbalanced data, the activity rate is so low that some methods may predict inactive status for the entire collection. Although the resulting accuracy rate is high, we miss all compounds that are truly active. Since one of the major goals of a QSAR study is to identify chemical structures that lead to active chemical reaction, QSAR modeling needs the ability to handle imbalanced data.

Significantly different chemical structures may cause the same biological activity. In other words, multiple mechanisms can lead to the same biological response,[2] and therefore, correct prediction depends on the ability to distinguish multiple regions of activity amidst an overwhelming excess of inactive structures. It is difficult to detect all the mechanisms, and most conventional methods may only recognize some of them. When using a multiple linear regression model, for example, we may detect at most one mechanism. In general, the number of mechanisms is unknown, and many popular QSAR methods fail because of this uncertainty. DT, for example, may ignore some active structures due to using a single descriptor as the splitting variable.[10] With these complexities induced by multiple mechanisms, the ability to detect multiple mechanisms is crucial for QSAR modeling.

No single modeling approach has been shown to be optimal for all QSAR studies. Moreover, some modeling approaches have been shown to be highly sensitive to small perturbations in their training data. For this reason, the method of ensembling has gained popularity in recent years.[7] The method of ensembling aggregates results from several individual models in an attempt to achieve substantial improvement over all individual models. Ensemble models can be designed in many different ways. Dietterich[11] points out that the performance of ensembles depends critically on three factors.

In this paper, we focus on the "family ensemble". Our choice of the word family indicates that all individual models used as input of the ensemble come from a common ancestor, i.e., a common data-mining algorithm. There are three factors

* Corresponding author. Phone: 16465415430. E-mail: jessie8015@hotmail.com.
† North Carolina State University.
‡ University of British Columbia.

10.1021/ci900080f CCC: $40.75
Published on Web 07/13/2009

for implementing a family ensemble: (a) the base learner, which is the data-mining algorithm used to create all individual models for the ensemble, examples are DT and KNN; (b) the selection of training data sets − the goal is to create an ensemble whose individual input models are as diverse as possible; and (c) the strategy for combining results from all individual input models, including specifying weights on the results of all base learners, e.g., the strategy of majority voting assigns equal weights to each learner. RF belongs to the class of family ensembles. It uses DT as the base learner, bootstrapping to select the training set, and makes ensemble prediction by a majority vote.

A recent study by Bruce et al.[12] compared many family ensembles for their effectiveness on balanced biological activity data. They concluded that several family ensembles are more accurate than individual methods, but a single decision tree remains competitive. Because imbalance is a common feature of QSAR data sets, we believe it is important to study the performance of family ensembles on imbalanced data sets. Moreover, we also believe that well-constructed family ensembles, which use an "unstable" base learner (able to obtain very different results from slightly different subsets of the training data set) and employ a flexible method to aggregate the results, can outperform the individual method on imbalanced data. Our goal in this paper is to propose a model-based ensembling method that provides good prediction on imbalanced data. Additionally, we motivate use of the F-measure as an appropriate assessment criterion for QSAR studies.

The construction for model-based ensembling is described in the Methods Section. We investigate the proposed ensemble method by studying the data sets used by Bruce et al.[12] plus two additional data sets obtained from the Molecular Libraries Screening Center Network[9] through PubChem.[8] Results for these data sets and discussions comparing performance are presented in the Results and Discussion Section. Finally, in the Summary Section, we review the advantages and limitations of the model-based ensembling method.

METHODS

The proposed method, model-based ensembling (MBEnsemble), is developed for QSAR classification problems that make predictions for binary designations of activity. MBEnsemble is able to automatically adjust the decision rule determining prediction that a compound is active according to the performance of base learners on training data sets. As will be demonstrated later in this paper, this feature is especially suitable for imbalanced data. All feature of MBEnsemble will be further discussed in this and subsequent subsections. To construct a family ensemble, it is necessary to carefully decide three factors: (a) the base learner for the ensemble; (b) the manipulation of the data set for each learner; and (c) the scheme to combine the results from all learners. Before introducing MBEnsemble, we first discuss some basic concepts used to develop MBEnsemble.

Probability Averaging. There are various schemes to aggregate multiple learners. Majority vote (MV) is a common choice, e.g., RF uses MV for classification problems. It jointly uses the learners by counting a vote from each learner, and the class that receives the largest number of votes is selected as the final decision. Suppose there are m independent learners in the ensemble and each learner has an accuracy rate of θ. The accuracy rate of the ensemble using MV is

$$\theta_{E,MV} = \begin{cases} \sum_{k>m/2}^{m} \binom{m}{k} \theta^k (1-\theta)^{m-k} & m \text{ is odd} \\ \sum_{k>m/2}^{m} \binom{m}{k} \theta^k (1-\theta)^{m-k} + \frac{1}{2}\binom{m}{m/2} \theta^{m/2} (1-\theta)^{m/2} & m \text{ is even} \end{cases}$$

While MV has received much attention, other schemes can be more effective. Using a data set on handwritten digit recognition, Kittler et al.[13] compared six combination schemes: sum, min, max, product, and median rules and MV. They concluded that the median rule outperformed the other five combination schemes. For data with binary designations of activity, probability averaging (PA), which averages over the predicted probabilities of being active obtained from all m learners, is a competitive alternative to the median rule. As the mean of probabilities, PA has benefits over the median rule and results in the following approximation of ensemble accuracy rate:

$$\theta_{E,PA} \approx \Pr(Z \leq z_\theta \sqrt{m})$$

where Z represents a random variable that follows the standard Gaussian distribution, $\Pr(Z \leq z)$ is the cumulative distribution function, and $\Pr(Z \leq z_\theta) = \theta$.

Figure 1 displays the gain in accuracy due to PA relative to the accuracy due to MV, namely $(\theta_{E,PA} - \theta_{E,MV})/\theta_{E,MV}$. PA can get more than 6% improvement in accuracy over MV if the base learner is more accurate than that of a learner that performs random selection, i.e., $\theta > 0.5$. The goal of ensembling is to achieve the best possible performance, and the base learner used in the ensemble is usually better than random guessing. Therefore, we choose PA to make decisions in MBEnsemble.

Threshold. The above accuracy rates, $\theta_{E,MV}$ and $\theta_{E,PA}$, assume that the base learners are independent of each other. In practice, it is difficult to ensure independence among all

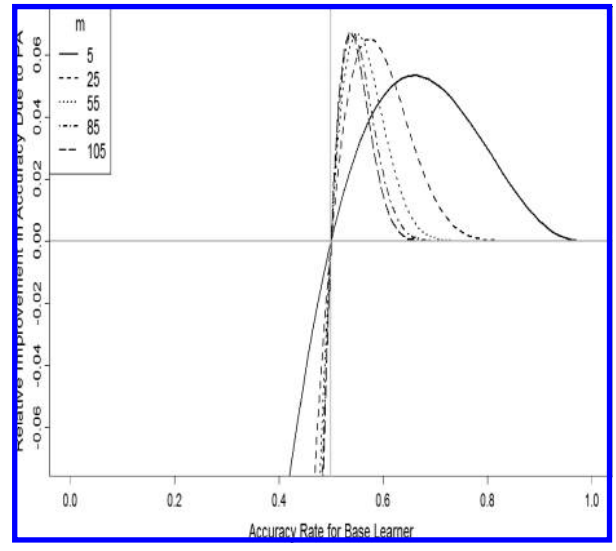

Figure 1. Improvement in accuracy rate due to PA relative to MV's accuracy rate, i.e., $(\theta_{E,PA} - \theta_{E,MV})/\theta_{E,MV}$. The improvement depends on m, the number of independent learners in the ensemble, and θ, the accuracy rate for base learners.

base learners. To relax the restriction of independence and have a further understanding of ensemble prediction based on PA, we generalize the assumptions as follows: (1) V_{Yi}, the estimated probability of being active from the i^{th} learner for the compound with true response Y ($0 \equiv$ inactive, $1 \equiv$ active), follows a distribution with mean μ_Y and standard deviation σ_Y; (2) the correlation between V_{Yi} and V_{Yj} for $i \neq j$ is ρ_Y; (3) V_{1i} and V_{0i} are independent of each other; and (4) the truly activity rate is known to be p. PA prediction is based on a preset threshold δ and

$$\bar{V} = \frac{1}{m}\sum_{i=1}^{m}\{V_{1i}I(Y=1) + V_{0i}I(Y=0)\}$$

where $I(\cdot)$ is the indicator function. If $\bar{V} > \delta$, then the compound is predicted to be active; otherwise, the compound is predicted to be inactive.

When m is large ($m = 100$, used in this paper, is typically considered large), $\bar{V}$ approximately follows a mixture-of-normals distribution. The components of this mixture are: a normal component with mean μ_1 and standard deviation $\sigma_1(1 + (m-1)\rho_1)^{1/2}$ (m with weight p)$^{1/2}$, and a normal component with mean μ_0 and standard deviation $\sigma_0(1 + (m-1)\rho_0)^{1/2}$ (m with weight $1-p$)$^{1/2}$. The above assumptions imply the accuracy rate of the ensemble using PA as

$$\theta_{PA} \approx p\Pr\left(Z > \frac{\delta-\mu_1}{\sigma_1}\sqrt{\frac{m}{1+(m-1)\rho_1}}\right) + (1-p)\Pr\left(Z \leq \frac{\delta-\mu_0}{\sigma_0}\sqrt{\frac{m}{1+(m-1)\rho_0}}\right)$$

Parameters μ_Y, σ_Y, and ρ_Y depend on the base learner chosen for the ensemble, the scheme for selecting a training data set for each base learner, and the richness of the data. On the other hand, the activity rate p depends only on the data. Once we decide the construction scheme of the ensemble for a given data set, then p, μ_Y, σ_Y, and ρ_Y become unchangeable, and so prediction quality is controlled only by adjusting δ. In the Results and Discussion Section, we will show the role of δ and how distributions of V_{1i} and V_{0i} affect the selection of δ.

Assessment Using the F-Measure. In learning from imbalanced data, accuracy rate is an inappropriate measure of performance. There are many alternative measures for performance evaluation. Misclassification cost,[14] F-measure,[15] and G-mean[16,17] are common choices to assess performance on imbalanced data. These measures are functions of the confusion matrix as shown in Table 1. Given the unit cost of a false negative (*FN*), c_1, the unit cost of a false positive (FP), c_0, and the total number of compounds, N, the misclassification cost, F-measure and G-Mean can be defined respectively as:

- Misclassification cost $= (c_1FN + c_0FP)/N$. All nonnegative values are possible, with 0 being ideal.
- G-Mean $= ((TP/(TP + FN))(TN/(TN + FP)))^{1/2} \equiv (a^+a^-)^{1/2}$. Values range from 0 to 1, 1 being ideal.
- F-measure $= ((1+\alpha)TP)/((1+\alpha)TP + FP + \alpha FN) = (1+\alpha)/[(TP+FP)/TP + \alpha(TP+FN)/TP]$, where $\alpha \geq 0$ is set by the user. Values range from 0 to 1, 1 being ideal.

According to the above definitions, the ratio of c_1 to c_0, $c = c_1/c_0$, has an obvious influence on the use of misclassification cost. If c is very large, then most attention will be paid to reduce *FN*, and the misclassification cost will lose the control of *FP* as well as its responsibility of assessment. Large c may lead the algorithm to predict actives for all compounds if the goal of the algorithm is to minimize the misclassification cost. In reality, the ratio c is large since the cost of misclassifying a rare active compound to be inactive is quite high. Although the 100% active prediction minimizes the misclassification cost, it does not provide valuable results for a QSAR study.

Table 1. Confusion Matrix

	predicted active class	predicted inactive class
truly active class	*TP*	*FN*
truly inactive class	*FP*	*TN*

The G-mean is a popular assessment measure that is typically used outside of QSAR studies.[16,17] Based on the proportion of truly active compounds that are correctly predicted (a^+) and the proportion of truly inactive compounds that are correctly predicted (a^-), the G-mean is high when both a^+ and a^- are high and the difference between a^+ and a^- is small. Consequently, the G-mean applies equal weights to correctly identifying actives and inactives. While this strategy is preferable to only monitoring the overall accuracy rate, it still is not entirely appropriate for QSAR goals. For QSAR studies, there is very little (likely no) interest in identifying inactive compounds, hence, a^- is not informative, and so an assessment measure based on a^- is not attractive. In fact, *TN* in Table 1 is of little use because correctly identifying inactives is not of primary value in QSAR studies. Hence, we argue that because both the misclassification cost and the G-mean directly involve *TN*, they are less desirable for assessing QSAR model effectiveness for binary outcomes in the presence of imbalanced classes.

In the spirit of misclassification cost, the F-measure uses α to control the numbers of *FN* and *FP*. When α approaches zero, the F-measure approaches a measure that is quite popular in the text-mining literature, namely the *precision*, where precision is defined as $TP/(TP + FP)$. Precision is exactly equivalent to the *hit rate* that is more commonly known in the QSAR community. When α approaches infinity, the F-measure approaches another popular measure called *recall*, where recall is defined as $TP/(TP + FN)$. Recall is the proportion of truly active compounds that are predicted to be active. Using the notation introduced for the G-mean, recall is exactly equivalent to a^+.

The F-measure is actually a weighted harmonic mean of precision and recall, and α is the weight for recall. Therefore, the F-measure takes values between 0 (indicating the worst performance) and 1 (indicating the best performance). A commonly used F-measure is F_1 that uses $\alpha = 1$ and has equal weight on recall and precision.

A G-mean based on precision and recall has also been proposed,[17] defined as $((TP/(TP + FN))(TP/(TP + FP)))^{1/2}$, where values range from 0 to 1, 1 being ideal. This measure enjoys many of the benefits of the F-measure but does not allow unequal weights to be applied to precision and recall. Although we use the equal-weight version of the F-measure for the remainder of this paper, we ascribe to the belief that there are studies for which unequal weights are appropriate and necessary, and hence, we find value in the F-measure.

To more clearly see difference and similarities between the six assessment measures accuracy (A), misclassification cost

(*MC*), G-mean based on accuracy rates (G_1), G-mean based on precision and recall (G_2), the equal-weight F-measure (F_1), and an unequal-weight F-measure (F_2), consider the following two confusion matrices (**A** and **B**), both ordered as described in Table 1 with $c_1 = 10$ and $c_0 = 1$:

A: $TP = 90$, $FN = 10$, $FP = 20$, $TN = 80$. Then $A = 0.85$, $MC = 0.60$, $G_1 = 0.85$, $G_2 = 0.86$, $F_1 = 0.86$, $F_2 = 0.87$.

B: $TP = 90$, FN = 10, $FP = 200$, $TN = 800$. Then $A = 0.81$, $MC = 0.27$, $G_1 = 0.85$, $G_2 = 0.53$, $F_1 = 0.46$, $F_2 = 0.55$.

Confusion matrix **B** is reflective of imbalance, and most would agree that the predictive model is far less effective in this case than in matrix **A**. However, G_1 rates these matrices equally while *A* assigns only marginal penalty to matrix **B**. Even worse, *MC* rates matrix **B** as better than matrix **A**. On the other hand, G_2, F_1, and F_2 all significantly penalize matrix *B*, albeit in varying amounts.

To provide further assistance in calibrating numerical values of the F-measure to other assessment measures, two addition confusion matrices (**C** and **D**) are considered below. These matrices fix the total number of truly active and truly inactive compounds to match the corresponding totals in confusion matrices **A** and **B** discussed above, but instead use a "random guess" approach to arbitrarily provide correct prediction for half of the true actives and correct predictions for half of the true inactives. Assessment measures on confusion matrices **C** and **D** are shown below:

C: $TP = 50$, $FN = 50$, $FP = 50$, $TN = 50$. Then $A = 0.50$, $MC = 2.75$, $G_1 = 0.50$, $G_2 = 0.50$, $F_1 = 0.50$, $F_2 = 0.50$.

D: $TP = 50$, $FN = 50$, $FP = 500$, $TN = 500$. Then $A = 0.50$, $MC = 0.91$, $G_1 = 0.50$, $G_2 = 0.21$, $F_1 = 0.15$, $F_2 = 0.20$.

Again we see that G_2, F_1, and F_2 significantly penalize matrix **D** but in varying amounts, and their values are much worse for these random guesses than for matrices **A** and **B** that result in more true positives and fewer false negatives and false positives.

Based on features of the misclassification cost, G-mean and F-measure, we choose F_1 as the performance assessment measure and use it as a tool to find the appropriate threshold δ. Other options for α can also be used, depending on the needs of the study. Results, as presented in the Results and Discussion Section, using other values of α are available upon request.

MBEnsemble. As mentioned in the beginning of the Methods Section, there are three factors determining the construction and performance of a family ensemble. For Factor (a), DT is probably the most desired learner.[12] There are several reasons for this choice: the DT algorithm is very scalable for binary classification and, hence, can work for very large data sets; DT has the ability to deal with collinear descriptors; DT is interpretable; and DT can give big changes in estimation as a result of small changes in the training data set, thus, leading to less correlated learners. Because of its valuable properties, Bruce et al.,[12] Svetnik et al.,[7] and Dietterich[11] all suggest using DT as the base learner for ensembling methods. Accordingly, MBEnsemble consists of 100 decision trees.

For Factor (b), 10-fold cross-validation is used (the result by Kohavi[18] implies that 10-fold cross-validation with decision tree works well), and only 70% of the descriptors are randomly selected for each training data set in each fold. When it comes to selection of base learners for an ensemble, there are two major factors that greatly impact performance: the correlation between any two models in the ensemble and the strength of individual models. It is known that the ensemble method works best if base learners are independent of each other.[6,18] Increase of the correlation decreases the strength of the ensemble. Increase of the strength of individual models increases the strength of the ensemble. Unfortunately, increasing the percentage of descriptors used to construct each individual tree increases both correlation and strength of individual trees. We conducted a trial on a simulated data set (with 1 000 observations, 100 variables, and 7.5% activity rate). Nine percentages were considered for how many descriptors to include in training: 10, 20, ..., 90%. The results are in favor of using 70% of the descriptors to construct individual trees for the ensemble.

Alternatively, other techniques could be employed for selecting base learners (indeed, we find the techniques used in RF to be desirable). The primary focus in this paper is combining the base learners by adjusting the threshold and using PA, and these procedures are applicable no matter how one chooses to create base learners.

For Factor (c), probability averaging is implemented in MBEnsemble because of its great appeal, as mentioned earlier. When using PA, the threshold δ becomes a tuning parameter that is used to control the prediction of being active. Different δs may be needed for different data, especially for imbalanced data. Furthermore, the ideal δ relies on the properties of the base learner, e.g., μ_1, σ_1, μ_0, and σ_0. It is difficult to decide a good δ before the analysis. Therefore, MBEnsemble is designed to automatically choose the optimal δ enroute to its analysis.

The procedure of MBEnsemble is listed as follows:

Loop A: for i in (1:10), do 10-fold cross validation.

Loop B: for j in (1:100), use 100 decision trees for analysis.

I. randomly select 70% of descriptors from the complete data matrix to get data $D_{i,j}$.

II. use the i^{th} fold of $D_{i,j}$ as a test set and the rest of $D_{i,j}$ as the training set.

III. run the decision tree on the training set to obtain the model $M_{i,j}$.

IV. use model $M_{i,j}$ to estimate the probabilities of being active for the training set $P'_{i,j}$.

V. use model $M_{i,j}$ to estimate the probabilities of being active for the test set $P_{i,j}$.

End Loop B.

- use the PA scheme to aggregate $P'_{i,j}$ and get

$$\bar{P}'_i = \frac{1}{100}\sum_{j=1}^{100} P'_{i,j}$$

for the prediction on all the folds except the i^{th} fold.

- find the optimal threshold δ_i that maximizes the value of F_1 (F-measure with $\alpha = 1$) based on $\bar{P}'_i$.
- use PA on $P_{i,j}$ to obtain

$$\bar{P}'_i = \frac{1}{100}\sum_{j=1}^{100} P'_{i,j}$$

and the optimal δ_i to make prediction on the i^{th} fold data.

End Loop A.

Table 2. Summary of Data Sets

data set	compound type	no. comp	no. actives[a]	active rate, in %[a]	no. descriptors
ACE	angiotensin converting enzyme	114	23	16	56
ACHE	acetyl-cholinesterase inhibitors	111	22	20	63
BZR	benzodiazepine receptor	163	29	18	75
COX2	cyclooxygenase-2 inhibitors	322	66	20	74
DHFR	dihydrogolate reductase inhibitors	397	79	20	70
GPB	glycogen phosphorylase b	66	13	20	70
THERM	thermolysin inhibitors	76	15	20	64
THR	thrombin inhibitors	88	18	20	66

[a] Binary activity was obtained by thresholding the continuous assay measurement at the 84th, 80th, or 82nd percentile.

The inner Loop B in MBEnsemble ensures multiplicity for the ensemble because it creates 100 different DT models. Combining those DT models supports the detection of multiple mechanisms. The outer Loop A of MBEnsemble controls cross validation and selection of control parameter δ. As such, the outer loop works to decrease the variance of ensemble estimation and resists overfitting the data.

A natural question concerns possible overfitting due to our search for optimal thresholds in Loop A. As explained in the MBEnsemble pseudoalgorithm, 10-fold cross validation is used inside Loop A. Consequently, 10 optimal thresholds are determined. Each threshold is the optimal choice for the subset of 90% of the compounds used to construct trees in Loop B. With this threshold and the trees constructed in Loop B, we make predictions on the remaining 10% of the compounds that are not used in the construction of trees. Prediction on this set is fair because the set is excluded from tree construction and search for the optimal threshold. Therefore, MBEnsemble is less prone to issues with overfitting.

With MBEnsemble, we do not need to specify δ beforehand and, hence, "optimal" thresholds can be identified through the analysis. Such properties will benefit the analysis on imbalanced data. Empirical results shown in the next section indicate how analyses on imbalanced data profit from MBEnsemble.

RESULTS AND DISCUSSION

Data. In this section, we will discuss results of eight small data sets from the study of Bruce et al.[12] as well as two larger assays obtained from PubChem.[8] The earlier discussion on the choice of δ will continue, and the importance of δ will be displayed through empirical results obtained from these data sets. Moreover, MBEnsemble results will be compared with results from RF (an ensemble method based on MV) and a single DT (using 0.5 as a threshold to distinguish between active and inactive compounds).

A summary of the eight small data sets studied by Bruce et al.[12] is shown in Table 2. The assay measurement for these original data sets is continuous and shows a uniform distribution. Bruce et al.[12] focused on balanced classification, so they created binary responses by thresholding the continuous assay response at the median. Our study focuses on classification in the presence of imbalanced class counts, so we applied thresholds other than the median. While we studied many thresholds, even those that resulted in activity rates as low as 10%, we only present results corresponding to a near 20% activity rate. Due to ties in assay values at the threshold, the actual activity rates fluctuated around 20%.

Table 3. Summary of Assays

assay	descriptor type	no. descriptors
	BN	24
AID364	PF	121
no. compounds = 3381	AP	395
no. actives = 49	FP	597
activity rate = 1.4%	CAP	1 578
	BN	24
AID371	PF	119
no. compounds = 3312	AP	382
no. actives = 278	FP	580
activity rate = 8.4%	CAP	1 487

Table 2 shows activity rates of the eight small data sets; only ACE and BZR do not have activity rate of 20%.

Bruce et al.[12] use two types of descriptors: 2.5D descriptors generated by Sutherland et al.[20] and linear fragment descriptors. In this paper, we focus on the 2.5D descriptor set. Among the eight data sets, GPB, THER, and THR are quite small. In these data sets, the number of descriptors nearly equals the number of compounds. Therefore, these three data sets have more challenges for QSAR modeling. Furthermore, GPB is believed to be the most difficult data set among the eight data sets because it has only 66 compounds, and the number of descriptors are greater than the number of compounds.

The two large assays are assay AID364 and AID371. Both assays are expected to experience modeling challenges. Assay AID364 is a cytotoxicity assay with a 1.4% activity rate; the data was downloaded from PubChem[8] on June 4, 2006. Assay AID371 is an assay of A549 lung tumor cell growth inhibition with a 8.4% activity rate; the data was downloaded from PubChem[8] on November 2, 2006. Because toxic reactions can occur in many different ways, multiple mechanisms are expected in both assays, and we expect difficulty detecting all the mechanisms.

There are a large number of different sets of molecular descriptors for quantitatively representing chemical structure. Nevertheless, there is no consensus of opinion on types of input descriptors for QSAR models because a descriptor can achieve success for some targets but fail for other targets. Since the choice of descriptor is target-dependent, we report results of five types of descriptors for the two PubChem assays studied in this paper. With the descriptor generation engine of PowerMV (Liu et al.),[21] five sets of descriptors were generated for each assay: weighted Burden numbers (BN), pharmacophores fingerprints (PF), atom pairs (AP), fragment pairs (FP), and Carhart atom pairs (CAP). Table 3 summarizes the two assays with different descriptor types.

Table 4. Average F_1 from Nine Replicate Runs of 10-Fold Cross Validations of Ensembling 100 DTs with PA and Varying Preset δ for the Small Data Sets[a]

	δ								
data set	0	0.1	0.2	0.3	0.4	0.5	0.6	δ_{opt}	F_1 (δ_{opt})
ACE	0.336	0.658	0.680	**0.696**	0.692	0.659	0.615	0.30	0.696
ACHE	0.331	**0.516**	0.505	0.494	0.468	0.453	0.384	0.07	0.522
BZR	0.302	**0.451**	0.445	0.413	0.362	0.352	0.298	0.17	0.465
COX2	0.340	0.517	0.524	**0.531**	0.503	0.484	0.427	0.28	0.540
DHFR	0.332	0.540	0.589	**0.607**	0.596	0.567	0.488	0.34	0.608
GPB	0.329	0.542	**0.566**	0.560	0.499	0.471	0.303	0.27	0.577
THERM	0.330	0.523	**0.548**	0.536	0.540	0.517	0.491	0.25	0.561
THR	0.340	0.444	0.482	0.504	**0.526**	0.522	0.451	0.45	0.535

[a] **Bold**: The highest F_1 that was achieved among the seven preset thresholds ($\delta = 0, 0.1, ..., 0.6$).

Results − Specification of δ. In the Data Section, we mentioned that the threshold δ is an important tuning parameter when using PA to make predictions. To show the importance of δ, we run a pedagogic ensemble that consists of 100 decision trees with 10-fold cross validations. Predictions of the pedagogic ensemble are based on PA with a preset threshold. The major difference between the pedagogic ensemble and MBEnsemble is that MBEnsemble determines and uses the optimal threshold δ_i on the i^{th} fold data, while the pedagogic ensemble uses a preset threshold on the complete data set. The results of F_1 (F-measure with $\alpha = 1$) for the pedagogic ensemble with varying preset threshold δ are reported in Tables 4 and 5. We actually report averages of nine replications for the small data sets (in Table 4) and averages of three replications for the large assays (in Table 5). The value in bold denotes the highest F_1 that was achieved among the seven preset thresholds ($\delta = 0, 0.1, ..., 0.6$) for the data set. Also shown are the optimal F_1 value F_1 (δ_{opt}) for this pedagogic study as well as the threshold δ_{opt} that achieves this optimum.

We first consider the results in Table 4. The table shows that the values of F_1 heavily depend on the choice of δ: (1) when $\delta = 0$, the value of F_1 is small because the number of false positives reaches its maximum, which is equal to the number of inactive compounds, and far exceeds the number of true positives, which is equal to the number of active compounds; (2) the optimal threshold δ_{opt} resulting in the highest value of F_1 is always less than 0.5 for all eight data sets; this confirms that using 0.5 as the threshold may not provide favorable performance for imbalanced data; (3) δ and F_1 are positively correlated if $\delta < \delta_{opt}$, while δ and F_1 are negatively correlated if $\delta > \delta_{opt}$, i.e., the relationship between δ and F_1 appears to be unimodal, thus, suggesting an algorithm aimed at determining optimum δ (such as MBEnsemble) has likelihood for success; and (4) data sets with similar activity rate p can have quite different values of δ_{opt}. The inherent features of imbalanced data and the definition of F-measure account for observations (1−3) but not for observation (4). Therefore, we focus our discussion on observation (4).

As mentioned in the subsection of Probability Averaging, the appropriate choice of δ relies on distributions of V_{1i} and V_{0i}. Figure 2 displays estimated densities of V_{1i} and V_{0i} as dashed and dotted curves for data sets ACE and ACHE and illustrates how the distributions of V_{1i} and V_{0i} affect the location of δ_{opt}. Both densities of V_{0i} for ACE and ACHE have exaggerated peaks around zero, and hence, most predictions on truly inactive compounds are correct when δ is far enough from zero. On the other hand, both densities of V_{1i} have two peaks. For ACE, the peak around one is much higher than the peak around zero. This allows correct predictions on the majority of truly active compounds when δ is far enough from one and zero, and in this case the optimal δ is $\delta_{opt} = 0.30$. The distribution of V_{1i} for ACHE is contrary to that for ACE. Because of the high peak of V_{1i} around zero, it is difficult for ACHE to make correct predictions on most truly active compounds if δ is far from zero. As a result, the value of δ_{opt} for ACE is greater than the value of δ_{opt} for ACHE (which equals 0.07).

The impact of δ can even be demonstrated for a single tree. Figure 3 shows a tree obtained from a subset of the ACE data set. By default, δ is set to 0.5, and this results in the predicted classes shown as the number listed for each leaf (terminal node) of the tree; one indicates that compounds falling the leaf predicted as active while zero indicates prediction as inactive. The numbers shown in parentheses for each leaf are the estimated probabilities of being active. Using $\delta_{opt} = 0.30$ as suggested by Table 4, we clearly see that one additional leaf (probability of 0.43) would predict compounds as active, thus, possibly increasing the chance of identifying additional true actives.

The importance of specifying δ is more obvious in Table 5, since AID364 and AID371 are both extremely imbalanced. For AID364, the values of F_1 in bold are greater than at least 147% of the values of F_1 using $\delta = 0.5$. For AID371, the values of F_1 are almost zero when we use $\delta \geq 0.3$. Also, F_1 is very sensitive to the value of δ for AID371 with input of some descriptors; for example, for descriptor type AP, F_1 is 0.207 when $\delta = 0.1$ and falls to 0.007 when $\delta = 0.2$. All these observations again indicate that using $\delta = 0.5$ may be dubious for extremely imbalanced data and the choice of δ determine prediction quality.

Results − Comparisons. By specifying a preset threshold before the analysis, the above pedagogic ensemble exhibits the role of δ for prediction on imbalanced data but is not preferred over the data-driven MBEnsemble. Table 4 shows that the optimal threshold δ_{opt} varies with different data sets despite these sets having approximately equal activity rates, and Table 5 implies that the value of δ_{opt} can change when using different molecular descriptors for the same assay. Therefore, it is difficult for the pedagogic ensemble to determine a reasonable δ only based on simple data properties such as the observed activity rate, the number of compounds, and the number of descriptors.

With MBEnsemble, we do not need to determine the threshold beforehand and optimal thresholds can be automatically determined through the performance of base learners on training data sets. As a result, MBEnsemble is a better candidate as a family ensemble on imbalanced data. Moreover, its incorporation of careful cross validation, to avoid model building and assessment using the same set, makes MBEnsemble resistant to overfitting. Next, we will show the comparison results of F_1 from MBEnsemble, RF[22] using ntree = 100, nodesize = 5, and default settings in R), a single DT[23] using default settings in R, and random guessing as described in the subsection Assessment Using the F-Measure. Our ChemModLab web site, http://eccr.stat.ncsu.edu/ChemModLab/Default.aspx, provides a computing platform for QSAR modeling based on different methods,

Table 5. Average F_1 from Three Replicate Runs of 10-Fold Cross Validations of Ensembling 100 DTs with PA and Varying Preset δ for the Large Assays[a]

		δ								
assay	descriptor	0	0.1	0.2	0.3	0.4	0.5	0.6	δ_{opt}	F_1 (δ_{opt})
AID364	BN	0.029	0.189	0.238	**0.284**	0.236	0.182	0.103	0.31	0.289
	PF	0.029	0.199	**0.213**	0.192	0.114	0.078	0	0.23	0.222
	AP	0.029	0.254	**0.262**	0.177	0.061	0.013	0	0.11	0.264
	FP	0.029	0.270	**0.378**	0.361	0.273	0.179	0.101	0.20	0.378
	CAP	0.029	0.241	**0.281**	0.261	0.239	0.191	0.160	0.17	0.302
AID371	BN	0.155	**0.227**	0.098	0.045	0.023	0.007	0.005	0.11	0.235
	PF	0.155	**0.194**	0.043	0	0	0	0	0.08	0.202
	AP	0.155	**0.207**	0.007	0	0	0	0	0.11	0.214
	FP	0.155	**0.226**	0.157	0.021	0.005	0	0	0.13	0.237
	CAP	0.155	**0.241**	0.211	0.045	0.012	0	0	0.13	0.256

[a] **Bold**: The highest F_1 that was achieved among the seven preset thresholds (δ = 0,0.1, ..., 0.6).

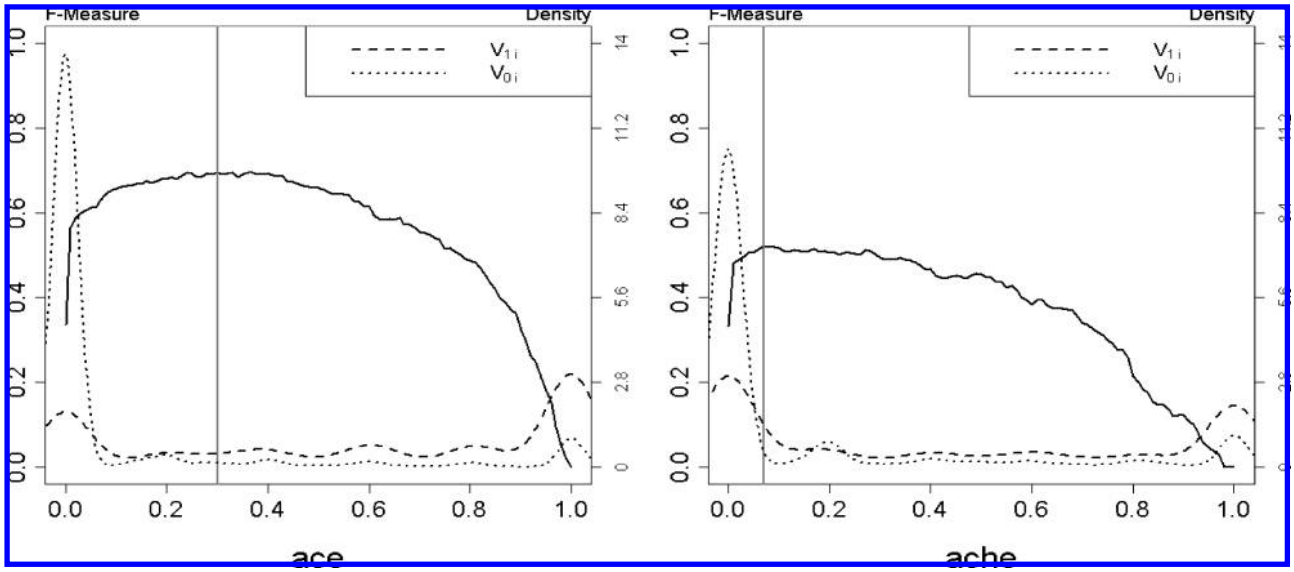

Figure 2. The left plot shows the results obtained from data set ACE, and the right plot shows the results obtained from data set ACHE. The dashed curve is the density plot for V_{1i}, the estimated probability of being active when the compound is truly active as reflected in the observed activity measurement. The dotted curve is the density plot for V_{0i}, the estimated probability of being active when the compound is truly inactive as reflected in the observed activity measurement. The solid curve shows the F-measure as a function of δ. The gray vertical line is a baseline displaying the location of δ_{opt}.

including RF and DT discussed here. Due to heavy computing, the MBEnsemble is not yet available on the ChemModLab web site.

Table 6 gives the average F_1 of nine replications for the eight small data sets. Using modeling approach as a four-level factor (with levels MBEnsemble, RF, DT, and random) and folds (from the 10-fold cross-validation exercise) as a second factor, an analysis of variance (ANOVA) was run with a subsequent application of Tukey's HSD to obtain multiple comparisons between modeling approaches. For each data set, the best statistically equivalent methods are in bold. Recall that a higher value of F_1 implies better performance of the model. The values of $F_1$are dependent on the data set — ACE has the most successful performance and BZR gets the least successful performance. With "optimal thresholds", MBEnsemble is one of the most effective methods for all eight data sets. For all of these small data sets, the difference between the F_1 of MBEnsemble and the minimum F_1 is between 10 and 65%. Even after accounting for uncertainty and variability. MBEnsemble is statistically better than RF for five of the eight data sets

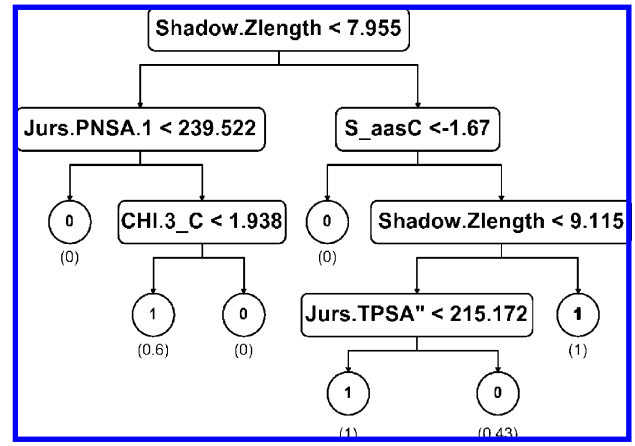

Figure 3. A tree constructed for data set ACE from randomly selecting 70% of the molecular descriptors to build the tree. The first number listed for each leaf is the prediction for that node, based on a default threshold of 0.5; 1 means active, 0 means inactive. The second number (shown in parentheses) is the estimated probability of being active.

Table 6. Average F_1 from Nine Replicate Runs of MBEnsemble, RF and DT[a]

data set	MBEnsemble	RF	DT	random
ACE	**0.703**	0.684	0.599	0.288
ACHE	**0.486**	0.474	0.442	0.284
BZR	**0.392**	(0.294)	**0.350**	0.262
COX2	**0.513**	0.446	0.450	0.291
DHFR	**0.591**	0.530	0.497	0.285
GPB	**0.513**	0.493	(0.310)	0.283
THERM	**0.580**	0.484	**0.521**	0.283
THR	**0.499**	(0.302)	**0.461**	0.290

[a] **Bold:** Best statistically equivalent methods after multiplicity adjustments; (): statistically equivalent to random guessing.

Table 7. Average F_1 from Three Replicate Runs of MBEnsemble, RF and DT[a]

assay	descriptor	MBEnsemble	RF	DT
	BN	**0.252**	0.249	0.182
AID364	PF	**0.223**	(0.052)	**0.137**
random:	AP	**0.239**	(0.052)	0.122
$F = 0.029$	FP	**0.316**	0.173	0.188
	CAP	**0.277**	0.115	**0.169**
	BN	**0.211**	0.243	0.030
AID371	PF	**0.201**	0.094	0
random	AP	**0.218**	(0.124)	0
$F = 0.144$	FP	**0.229**	(0.158)	0.031
	CAP	**0.255**	(0.122)	0.023

[a] **Bold:** Best statistically equivalent methods after multiplicity adjustments; (): methods statistically equivalent to random guessing.

(BZR, COX2, DHFR, THERM, and THR) and statistically equivalent to RF for the other three.

Surprisingly, RF does not gain improvement over DT for all data sets. DT has higher value of F_1 on four out of eight data sets, but its F_1 is not statistically higher than the F_1 of RF except for data set THR. The last column in Table 6 gives the performance for random guessing, which is used as the baseline method for model assessment. After accounting for variation, RF is equivalent to random guessing for BZR and THR, and DT is equivalent to random guessing for GPB. These results are contrary to the results obtained by Bruce et al.[12] They showed that ensembles (including RF) were superior methods and preferred to DT with respect to performance on balanced data sets. The data sets for Table 6 are all imbalanced. The disagreement between the balanced data sets and imbalanced data sets implies that it is possible for both RF and DT to fail in modeling imbalanced data sets. Next, we continue assessment for the three methods and inspect the performance of RF on extremely imbalanced data sets.

Table 7 compares F_1 of MBEnsemble, RF and DT on the two PubChem assays, which both have relatively low activity rate. Because of the relatively large number of compounds in these assays and the low activity rates, observed F_1 values tend to be rather small. For example, a method that results in perfect recall ($TP = 49$, $FN = 0$) and a hit rate ten times better than random guessing ($FP = 301$) for AID364 still results in the very low $F_1 = 0.25$. So, while the numbers in Table 7 are much smaller than those of Table 6, it should not be assumed that all results in Table 7 are unsuccessful attempts for prediction of these assays. In fact, several models provide very effective prediction for these PubChem assays.

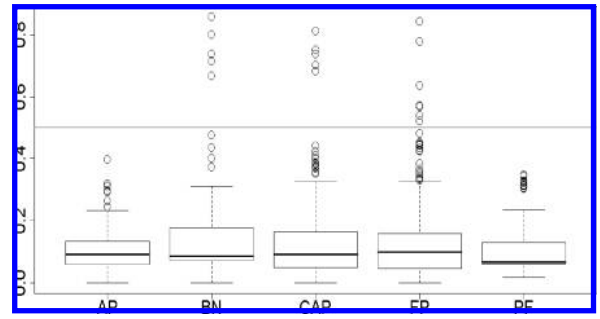

Figure 4. Box plots of DT-estimated probability of being active when the compound is truly active as reflected in the observed activity measurement (V_{1i}) for AID371.

Table 7 shows that MBEnsemble has stable performance on those assays and achieves the best performance. While both MBEnsemble and RF aggregate the results of 100 trees, the F_1 of MBEnsemble is between 0.201 and 0.316 and the F_1 of RF can vary a lot when using different molecular descriptors on the same assay. For example, F_1 for RF can increase 379% when changing molecular descriptors from PF (or AP) to BN on AID364. From Table 7, MBEnsemble provides dramatic improvement over RF, except when using BN descriptors. Statistical tests show that the difference in F_1 between MBEnsemble and RF is not significant for BN descriptors. Moreover, similar to Table 6, RF is not necessarily better than the DT in Table 7 and is statistically equivalent to or worse than random guessing for many cases. So Table 7 confirms possibly poor results when using RF on imbalanced data as studied in the paper.

Additionally, the performance of DT on AID371 is not comparable to any of the two ensemble methods, or even to random guessing. DT in this comparison uses 0.5 as the threshold. As mentioned in the earlier sections, $\delta = 0.5$ may not be an appropriate threshold for imbalanced data. Figure 4 illustrates why the F_1 of DT is almost zero on AID371, as revealed through estimated probabilities of being active when the compound is truly active (V_{1i}). For AP and PF, all DT-estimated V_{1i} are lower than 0.5, and this results in $TP = 0$ and, hence, $F_1 = 0$. For BN, FP, and CAP, the mean of DT-estimated V_{1i} is far below 0.5, and only a few "outliers" are greater than 0.5. So the value of TP is much smaller than the value of FN, and this results in low value of F_1 for BN, FP, and CAP.

We conclude that MBEnsemble outperforms the other studied methods for both the small data sets having approximately 20% activity rate as well as the larger PubChem assays, having less than 10% activity rate.

SUMMARY

This paper introduces an ensemble method, MBEnsemble, for building QSAR models. MBEnsemble selects a threshold based on the behaviors of its base learners on training sets rather than using a preset threshold in the decision rule for declaring that a compound is active. Imbalanced data benefits from this ensemble approach that allows flexibility with regard to thresholds. According to eight small data sets and two larger PubChem assays, MBEnsemble is the best of the studied methods on imbalanced data, even in the presence of multiple mechanisms within the PubChem assays.

The F-measure is used as the primary measure of assessment due to its relevance for awarding methods that correctly

identify actives and avoid faulty decisions. Comparisons to other assessment measures show the benefits of the F-measure for QSAR studies. This F-measure comparison shows that MBEnsemble is at least as good as, and often better than, RF and DT.

MBEnsemble is not perfect. It is computationally intensive yet does not always provide statistically significant improvements over the computationally attractive DT and RF for some data sets. Nevertheless, despite the current limitations, MBEnsemble provides stable performance on imbalanced data in the presence of multiple mechanisms. Empirical results also confirmed it is not suitable to use MV or a preset threshold for classification and prediction in the presence of the imbalanced data studied in this paper. Clearly, therefore, MBEnsemble is a powerful tool for developing QSAR models.

More importantly, some effective and essential components implemented in MBEnsemble (e.g., determining optimal thresholds and use of probability averaging) are directly transportable to other base learners, thus, allowing much broader application and potential impact. For example, the KNN algorithm can be made equally scalable as a decision tree for binary classification, but KNN has the additional benefit of flexible decision surfaces instead of the hyper-rectilinear decision surfaces implemented by decision trees. KNN could be used as the basis of an ensemble approach that incorporates probability averaging and selection of the number of neighbors according to F-measure optimization. This and other ensemble approaches will be the subject of future investigations.

ACKNOWLEDGMENT

This work was supported by the National Institutes of Health through NIH Roadmap for Medical Research, Grant 1 P20 HG003900-01. We thank Atina D. Brooks from North Carolina State University and S. Stanley Young from the National Institute of Statistical Sciences for early contributions to this work.

REFERENCES AND NOTES

(1) McFarland, J. W.; Gans, D. J. On the Significance of Clusters in the Graphical Display of Structure-Activity Data. *J. Med. Chem.* **1986**, *29*, 505–514.
(2) Rusinko, A.; Farmen, M. W.; Lambert, C. G.; Brown, P. L.; Young, S. S. Analysis of a Large Structure/Biological Activity Data Set Using Recursive Partitioning. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 1017–1026.
(3) Kauffman, G. W.; Jurs, P. C. QSAR and K-Nearest Neighbor Classification Analysis of Selective Cyclooxygenase-2 Inhibitors Using Topologically Based Numerical Descriptors. *J. Chem. Inf. Comput. Sci.* **2001**, *41*, 1553–1560.
(4) Doniger, S.; Hofmann, T.; Yeh, J. Predicting CNS Permeability of Drug Molecules: Comparison of Neural Network and Support Vector Machine Algorithms. *J. Comput. Biol.* **2002**, *9*, 849–864.
(5) Aoyama, T.; Suzuki, Y.; Ichikawa, H. Neural Networks Applied to Quantitative Structure-Activity Relationships. *J. Med. Chem.* **1990**, *33*, 2583–2590.
(6) Breiman, L. Random Forests. *Mach. Learn.* **2001**, *45*, 5–32.
(7) Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J. C.; Sheridan, R. P.; Feuston, B. R. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1947–1958.
(8) PubChem. BioActivity Services. http://pubchem.ncbi.nlm.nih.gov/assay (accessed June 4, 2006 for AID364, November 2, 2006 for AID371).
(9) Molecular Libraries Screening Centers Network. http://mli.nih.gov/mli/mlscn (accessed June 1, 2006).
(10) Zhang, K. Statistical Analysis of Compounds Using OBSTree and Compound Mixtures Using Nonlinear Models. *Electronic Theses and Dissertations at North Carolina State University Library*, 2006. http://www.lib.ncsu. edu/etd (accessed April 24, 2008).
(11) Dietterich, T. G. Ensemble Methods in Machine Learning. *Lect. Notes Comput. Sci.* **2000**, *1857*, 1–15.
(12) Bruce, C. L.; Melville, J. L.; Pickett, S. D.; Hirst, J. D. Contemporary QSAR Classifiers Compared. *J. Chem. Inf. Model.* **2007**, *47*, 219–227.
(13) Kittler, J.; Hatef, M.; Duin, R. P. W.; Matas, J. On Combining Classifiers. *IEEE Trans. Pattern Anal. Mach. Intell.* 1998, *20*(3), 226−239.
(14) Pazzani, M.; Merz, C.; Murphy, P.; Ali, K.; Hume, T.; Brunk, C. *Reducing Misclassification Costs*, Proceedings of the 11th International Conference on Machine Learning, Morgan Kaufmann: San Francisco, CA 1994.
(15) Rijsbergen, V. In *Information Retrieval*, 2nd ed.; Butterworth-Heinemann: London, U.K., 1979.
(16) Kubat, M.; Matwin, S. *Addressing the Curse of Imbalanced Data Sets: One-sided Sampling*. Proceedings of the 14th International Conference on Machine Learning, Nashville, TN, 1997; Morgan Kaufmann.
(17) Chen, C.; Liaw, A.; Breiman, L. *Using Random Forest to Learn Imbalanced Data*. Technical Reports for Department of Statistics at University of California, 2004, Berkeley, CA. http://www.stat.berkeley.edu/tech-reports/666.pdf (accessed April 26, 2009).
(18) Kohavi, R. *A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection*. Proceedings of the 14th International Joint Conference on Artificial Intelligence. San Francisco, CA, 1995; Morgan Kaufmann.
(19) Tan, P. N.; Steinbach, M.; Kumar, V. In *Introduction to Data Mining*; Addison Wesley: Boston, MA, 2005.
(20) Sutherland, J. J.; O'Brien, L. A.; Weaver, D. F. A Comparison of Methods for Modeling Quantitative Structure - Activity Relationships. *J. Med. Chem.* **2004**, *47*, 5541–5554.
(21) Liu, K.; Feng, J.; Young, S. S. PowerMV: A Software Environment for Molecular Viewing, Descriptor Generation, Data Analysis and Hit Evaluation. *J. Chem. Inf. Model.* **2005**, *45* (2), 515–522.
(22) Liaw, A.; Wiener, M. *R Package Random Forest*, Version 4.15−18; R Foundation for Statistical Computing: Vienna, Austria, 2006.
(23) Ripley, B. D. *R Package Tree*, Version 1.0−24; R Foundation for Statistical Computing: Vienna, Austria, 2006.

CI900080F