Comparative Study of Machine-Learning and Chemometric Tools for Analysis of In-Vivo High-Throughput Screening Data

Kirk Simmons,*,† John Kinney,‡ Aaron Owens,§ Dan Kleier,|| Karen Bloch,§ Dave Argentar,⊥ Alicia Walsh,§ and Ganesh Vaidyanathan#

Simmons Consulting, 52 Windybush Way, Titusville, New Jersey 08560, DuPont Stine Haskell Research Laboratories, 1090 Elkton Road, Newark, Delaware 19711 DuPont Engineering Research and Technology, POB 80249, Wilmington, Delaware 19880-0249, Drexel University, 3141 Chestnut Street, Philadelphia, Pennsylvania 19104, Sun Edge, LLC, 147 Tuckahoe Lane, Bear, Delaware 19701, and, Quantum Leap Innovations, 3 Innovation Way, Suite 100, Newark, Delaware 19711

Received April 24, 2008

High-throughput screening (HTS) has become a central tool of many pharmaceutical and crop-protection discovery operations. If HTS screening is carried out at the level of the intact organism, as is commonly done in crop protection, this strategy has the potential of uncovering a completely new mechanism of actions. The challenge in running a cost-effective HTS operation is to identify ways in which to improve the overall success rate in discovering new biologically active compounds. To this end, we describe our efforts directed at making full use of the data stream arising from HTS. This paper describes a comparative study in which several machine learning and chemometric methodologies were used to develop classifiers on the same data sets derived from in vivo HTS campaigns and their predictive performances compared in terms of false negative and false positive error profiles.

INTRODUCTION

High-Throughput Screening (HTS) is an integral component of many pharmaceutical, animal health, and crop-protection discovery operations. However, the implementation of HTS can differ significantly between these industries in that crop-protection screening libraries can be evaluated directly against the pest species of interest at very early stages in the discovery process. This is often accomplished using miniaturized 96-well plate-based assays evaluating the affect of a compound directly on the viability of the intact insect, weed, or fungal pathogen. A potentially significant value of in vivo HTS screening is that completely unexpected or novel mechanisms of action may be discovered as a result of a screening campaign. However, HTS screening requires testing large numbers of compounds in order to produce the required number of hit and lead compounds for a discovery effort, a common theme in all of the industries. Therefore, HTS screening is a logical place to apply data mining and predictive modeling in order to enhance the overall success rate by potentially using such models to identify chemical structures more likely to be active once screened.

Over the years the performance of a number of data analytic methods has been evaluated by developing quantitative structure−activity relationships (QSAR) between chemicals and their associated biological activity.[1–8] Many of these earlier reports only focused on a single methodology, often using an in vitro biological end point unique to the paper, thus making it difficult to directly compare the effectiveness of each method. We were interested in a side-by-side comparison of several common analytic methods using the same or similar data sets in order to be able to evaluate the relative strength and weakness of each of the statistical techniques. Our intention was to ultimately develop and deploy into our compound acquisition process predictive models derived from all of our historical HTS screening results using the best of the modeling techniques. The present study is a logical extension of earlier work in which the effectiveness of various chemical descriptors were studied across common data sets[9] using a decision tree classifier to construct the models. Comparative studies have appeared recently in which several analytic methods were evaluated against common data sets in an attempt to provide a rational means of selecting a particular method given the analysis task at hand.[10–14] These reports have prompted us to report some of our earlier work in this area.[15]

METHODS

Biology. Screening results from a fungicide HTS bioassay were selected as the end point for the initial comparative study because, in general, we have found fungicide activity to be a challenging end point to predict among the common crop-protection assays (insecticidal, herbicidal, and fungicidal activity). Additionally, at the time of this study, there was strong interest in discovering additional fungicide leads from screening, and we hoped to be able to apply developed models to compound acquisitions targeted at enhancing the hit rates for our HTS process. Compounds were initially screened in the fungicide bioassay at a concentration of 17 μM against the Wheat Glume Blotch pathogen

* Corresponding author e-mail: KirkASimmons@gmail.com.
† Simmons Consulting.
‡ DuPont Stine Haskell Research Laboratories.
§ DuPont Engineering Research and Technology.
|| Drexel University.
⊥ Sun Edge, LLC.
Quantum Leap Innovations.

10.1021/ci800142d CCC: $40.75
Published on Web 08/06/2008

Table 1. Initial Data Sets for Side-by-Side Comparisons

data set	size	% active	train	test
A	1735	50%	1480	255
B	2656	33%	2271	385
C	4350	20%	3729	621

Phaeosphaeria nodorum (Ascomycete order) in an agar-based assay. Compounds that were active were advanced into higher level foliar-based screens to further explore the biological activity that was observed. Compounds deemed "inactive" for our studies were those producing less than 10% inhibition at 17 μM in the agar-based pathogen assay, and compounds deemed "active" were those that confirmed as active in advanced foliar-based screens at application rates at or below 200 mg/L applied concentration. The HTS screening results were subjected to a triage in which duplicate structures and those of unknown or questionable structure were removed. These conditions returned a fungicide data set consisting of ca. 629,000 structures of which ca. 2,600 were active (overall 0.42% active). The analytic methods being evaluated in this study have varying abilities to handle data sets of the size arising from HTS. In order to evaluate all of them side-by-side and assuming the results from modeling smaller data sets would be predictive of the results modeling the HTS data set, smaller data sets were constructed for analysis by random sampling from the fungicide HTS data set (Table 1).

Chemistry. All of the structures in the HTS data set were preprocessed using Pipeline Pilot[16] to standardize structural representations (e.g., -[N+](=O)[O-] => -N(=O)=O) and the structures output as a MDL connection table.[17] The 2D structures were converted to 3D using Concord, and the structures were output as a Sybyl mol2 connection table.[18] There are a large number of available chemical descriptors that could be computed from a connection table.[19] Earlier work[9] described a comparative study of the effectiveness at discriminating biological activity of some of the more common chemical descriptors using recursive partitioning modeling. Among those studied atom-pair descriptors[20] were found to be quite effective in modeling, and so the structures in our HTS data set were used to compute 3D atom-pair descriptors in which the structure is represented by all of the <atom type - distance - atom type> combinations in the structure. Our implementation of the atom-pair descriptors, coded in the C programming language,[21] collapses the ca. 25 Sybyl mol2 atom types into 10 atom types and maps the interatomic distances between them into 1-Angstrom distance bins (distances ranging from 1 to 15 Angstroms). The final descriptors consisted of 825 numerical values representing 55 possible atom-type pairs mapped to 15 distance ranges.

Model Development/Assessment. kNN (k Nearest Neighbors) analysis, logistic regression, and linear discriminant analysis were implemented using SAS software.[22] PLS analysis was performed using the PLS Toolbox in Matlab.[23] The development of Neural Networks[24] and InfoEvolve[25] models used DuPont proprietary software. Decision tree-based analyses were conducted using FIRM[26] (formal inference-based recursive modeling), OC1[27] (Oblique Classifier), and an implementation of C4.5[28] in OC1. CART models were developed using CART Pro 6 from Salford Systems.[29] Ensemble FIRM modeling was implemented using a proprietary routine coded in FORTRAN interfacing to FIRM.

Each of the three previously described data sets was randomly divided (85:15) into training and testing subsets for the purposes of model development. Models and the associated techniques were evaluated on the performance of predicting the test set, following model development on the training set. Predictive performance was assessed strictly

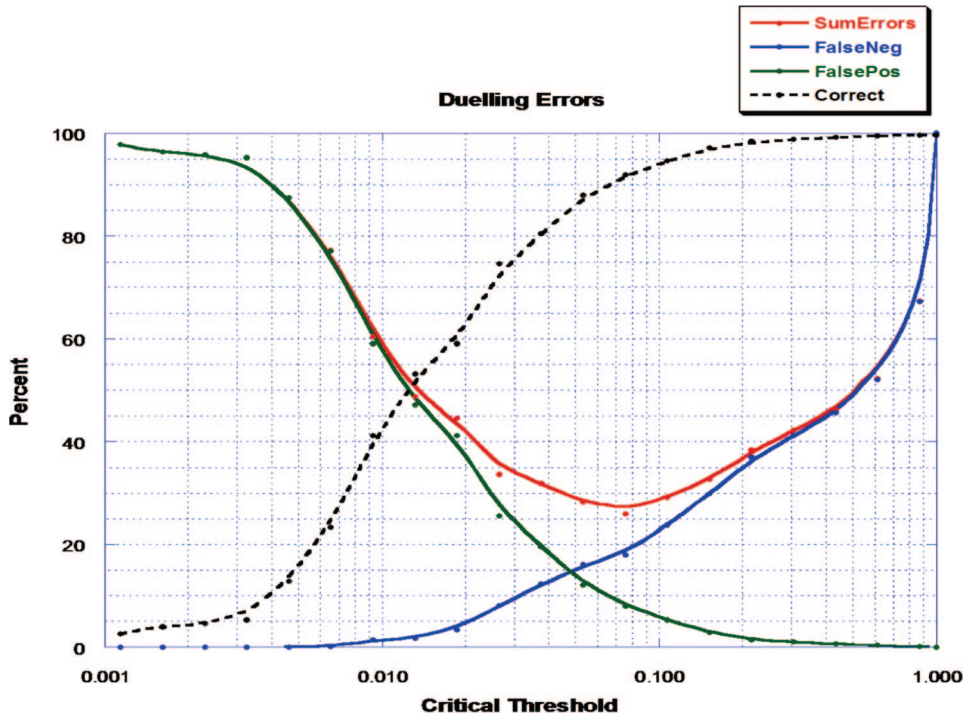

Figure 1. Behavior of the sum of False Negative and False Positive errors. A larger critical threshold implies a more stringent test for predicting a positive result (i.e., biologically active). A smaller critical threshold implies a more lenient test for activity.

Table 2. Prediction Summaries[a]

analytic method		data set	comparative study - test set performance FalseNeg	FalsePos	SumError	TruePos	TrueNeg
InfoEvolve		C (4:1)	15.4%	9.6%	25.0%	84.6%	90.4%
C4.5		C (4:1)	42.6%	2.2%	44.8%	57.4%	97.8%
		B (2:1)	24.0%	10.5%	34.5%	76.0%	89.5%
		A (1:1)	25.4%	20.8%	46.2%	74.6%	79.2%
CART		C (4:1)	27.1%	8.5%	35.6%	72.9%	91.5%
Oblique Classifier		C (4:1)	28.7%	6.3%	35.0%	71.3%	93.7%
		B (2:1)	23.3%	8.2%	31.5%	76.7%	91.8%
		A (1:1)	13.8%	18.4%	32.2%	86.2%	81.6%
kNN Analysis	k=1	C (4:1)	20.2%	6.7%	26.9%	79.8%	93.3%
	k=2	C (4:1)	14.7%	12.4%	27.1%	85.3%	87.6%
	k=3	C (4:1)	10.9%	18.3%	29.1%	89.2%	81.7%
	k=4	C (4:1)	8.5%	25.6%	34.1%	91.5%	74.4%
Logistic Regression		C (4:1)	20.0%	11.4%	31.4%	80.0%	88.6%
		A (1:1)	16.2%	12.0%	28.2%	83.8%	88.0%
Linear Discriminant		C (4:1)	22.5%	7.3%	29.8%	77.5%	92.7%
PLS Analysis		A (1:1)	15.4%	15.2%	30.6%	84.6%	84.8%
Neural Networks		A (1:1)	12.3%	19.2%	31.5%	87.7%	80.8%
FIRM		C (4:1)	19.4%	11.6%	31.0%	80.6%	88.4%
		B (2:1)	19.4%	25.4%	44.8%	80.6%	74.6%
		A (1:1)	20.0%	20.8%	40.8%	80.0%	79.2%
Ensemble FIRM		C (4:1)	19.4%	5.3%	24.7%	80.6%	94.7%
global average			20.0%	13.1%	33.1%	80.0%	86.9%

[a] False negatives are actually actives. Consequently, the true positive rate plus the false negative rate is logically expected to sum to 100%. Similarly, the true negative rate plus the false positive rate is expected to sum to 100%.

Table 3. Minimum SumError (False Pos + FalseNeg) by Data Set by Method

analytic method	comparative study - test set performance SUM of (FalsePos + FalseNeg) errors C (4:1)	B (2:1)	A (1:1)	Min Sum Error
Ensemble FIRM	24.7%			24.7%
InfoEvolve	25.0%			25.0%
kNN Analysis (k = 1)+	26.9%			26.9%
kNN Analysis (k = 2)	27.1%			27.1%
Logistic Regression	31.4%		28.2%	28.2%
kNN Analysis (k = 3)	29.1%			29.1%
Linear Discriminant	29.8%			29.8%
PLS Analysis			30.6%	30.6%
FIRM	31.0%	44.8%	40.8%	31.0%
Oblique Classifier	35.0%	31.5%	32.2%	31.5%
Neural Networks			31.5%	31.5%
kNN Analysis (k = 4)	34.1%			34.1%
C4.5	44.8%	34.5%	46.2%	34.5%
CART	35.6%			35.6%

based upon the sum of false positive and false negative errors, a value that reaches a minimum with classifier threshold (Figure 1). The choice of the threshold very much affects the overall error rate for the classifier. For example, if the threshold is set to 1.00, then essentially all compounds are forecast as inactive, and if the threshold is set to 0.00, then all compounds are forecast as active. Neither extreme is very useful, and clearly the optimal choice falls somewhere between the extremes. The errors that we report are the False Positive and False Negative errors for the classifier at the optimal threshold which is determined by the machine learning algorithm. We use Figure 1 to remind the reader that in modeling highly unbalanced data sets like those arising from HTS (hit-rates typically under 1%), one cannot assess a model performance simply based on overall accuracy (the black dashed line labeled "correct"). In the worst-case scenario, the classifier simply predicts all records to have the outcome of the majority class (in HTS data sets, inactive) and misses entirely the minority class (in HTS data sets, active) yet can still achieve high overall accuracy. The ultimate classifier needs to have good predictive capabilities for both classes. The True Positive rate reflects a classifier's ability to correctly predict truly active structures in a collection, while the False Positive error reflects how many truly inactive compounds are, in fact, falsely predicted as active. Since the classifiers were ultimately being deployed to support compound acquisition efforts for HTS screening, it is desirable for the False Positive error to be minimized while the True Positive rate is maximized. For example, if a classifier was perfect at predicting active structures (True Positive rate = 100%) but possessed a False Positive rate of 20%, then forecasting a compound collection of 1,000,000 structures would correctly score all of the true actives but would also score 200,000 inactive structures incorrectly as active requiring HTS evaluation of 200,000+ compounds to identify the active structures! For this reason, we chose to assess the models based on the sum of the two competing errors (i.e., false negative error plus false positive error), since both are important.

kNN Models. kNN models were developed using the SAS PROC DISCRIM procedure, specifying the NPAR method. The distance metric was Euclidean distance, and models were developed specifying $k = 1$, 2, 3, or 4 nearest neighbors. No maximum distance constraint was applied so $k = 1$ corresponds to the single closest neighbor regardless of distance, $k = 2$ the two closest, etc. While perhaps not ideal, not specifying a maximum distance constraint assures all compounds are predicted by the modeling exercise.

Logistic Regression Models. Logistic regression models were developed using the SAS PROC LOGISTIC procedure with stepwise variable selection.

Linear Discriminant Models. Linear discriminant models were developed using the SAS PROC STEPDISC procedure with stepwise variable selection.

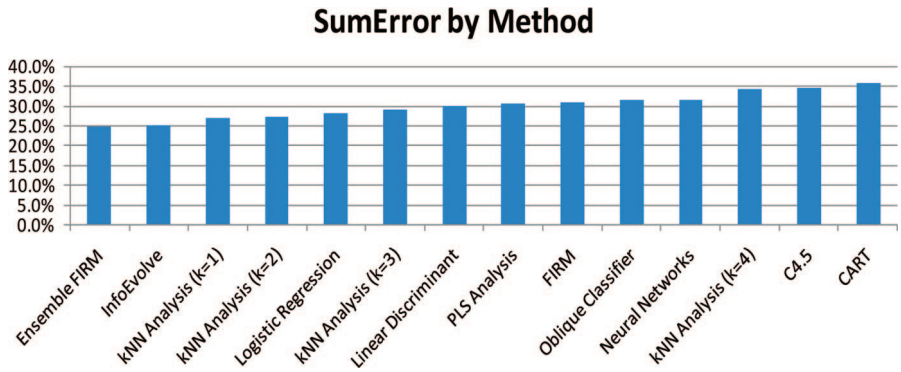

Figure 2. SumError% vs Modeling Method - best results.

Table 4. Classifier Performance Using the Entire HTS WGB Data Set

method	False Neg	False Pos	Sum Error	number of models
Fungicide/3d Atom Pairs				
Ensemble FIRM	19.9%	3.6%	23.5%	115
InfoEvolve	19.6%	5.2%	24.8%	150
Oblique Classifier	26.0%	9.1%	35.1%	1
Neural Networks				1
FIRM	19.9%	23.2%	43.1%	1
Fungicide/MolconnZ				
Ensemble FIRM	20.0%	5.1%	25.1%	55
InfoEvolve	20.1%	7.2%	27.3%	230
Oblique Classifier	20.6%	12.8%	33.4%	100
Neural Networks	20.0%	13.9%	33.9%	10
FIRM	20.0%	14.4%	34.4%	1

PLS Models. PLS models were developed using PLS Toolbox for Matlab.

Neural Network Models. Neural network models were developed stepwise by first randomly selecting 10% the training data set as a validation set. During training of the neural network the validation set was used to decide the optimal number of hidden units and to determine an error limit for terminating training. Using these conditions the neural network was then retrained using the entire training data set.

Oblique Decision Tree Models. Oblique classifiers search linear combinations of descriptors for splitting the data set and so offer the advantage of potentially simpler yet more accurate decision trees than parallel axis methods such as C4.5 and FIRM. However, they tend to be significantly more computationally intensive. The OC1 algorithm was used to induce a set of oblique decision trees using 10-fold cross-validation during training to select the final tree.

C4.5 Decision Tree Models. C4.5 decision tree models were developed using an implementation of C4.5 within OC1. Final models were selected using a 10-fold cross-validation. The reader is referred to Quinlan's description[28] of C4.5 for a discussion of the methodology.

CART Decision Tree Models. CART decision tree models were developed using CART Pro (version 6) and the Gini splitting criteria. The final model was selected based upon the results of a 10-fold cross-validation. The test data were then forecast using the final model.

FIRM Decision Tree Models. FIRM treats continuous descriptors by binning them initially into 10 approximately equally populated intervals. The routine analyzes the data set deciding which descriptor and which split cardinality is optimal at each node in the tree by systematically and recursively collapsing adjacent descriptor bins when they are not significantly different with respect to the predicted class. FIRM uses a chi-squared test (if class end point is nominal) or a *t* test (if the class end point is continuous) to decide the cardinality of the optimal split. The user can specify the minimum significance required during these tests as well as the overall significance of the variable choice. The minimum significances were set to a p-tail = 0.05 for all model development. Statistical test results are corrected using the Bonferroni adjustment to reflect the number of descriptors considered at each split. In addition FIRM allows the user to specify the smallest size node that is allowed to be further split. Tree induction ceases when statistical tests for descriptor selection exceed statistical thresholds, when the node size is below the splitting threshold specified by the user, or when the node is too homogeneous to split further.

Ensemble Decision Tree Models. Our approach to ensemble decision tree modeling involved wrapping FIRM with a FORTRAN procedure which managed data set sampling and model averaging while basically invoking FIRM for decision tree induction. Our sampling strategy consisted of resampling a data set, by selecting all of the actives in a data set and appending an approximately equal number of inactive structures randomly selected from the data set. Across resampling iterations the active structures are always reused, while the inactive structures are not. Thus data set C, which contains four inactive structures for each active one, was resampled to produce four smaller subsets consisting of a 1:1 inactive/active ratio. FIRM models were developed for each of these subsets, and the predictions were averaged across the four models.

InfoEvolve Models. The InfoEvolve method uses the same sampling strategy described for the ensemble decision tree method. InfoEvolve first analyzes each of the data subsets separately in order to identify the most information rich combinations of possible input variables for that data subset. All possible combinations of inputs are represented in a gene pool in which each bit in the gene represents each possible input. The global information content for the data set is used to drive the genetic algorithm based optimization. Once the evolution is completed there exists a pool of genes that represent optimal combinations of inputs which are most information rich for that data subset. Analysis of the frequency of occurrence of the inputs across this gene pool is used to select the final inputs considered in modeling each data subset. Once the optimal inputs are selected a second genetic algorithm-based optimization step occurs in which the root-mean-square (rms) error between predicted and

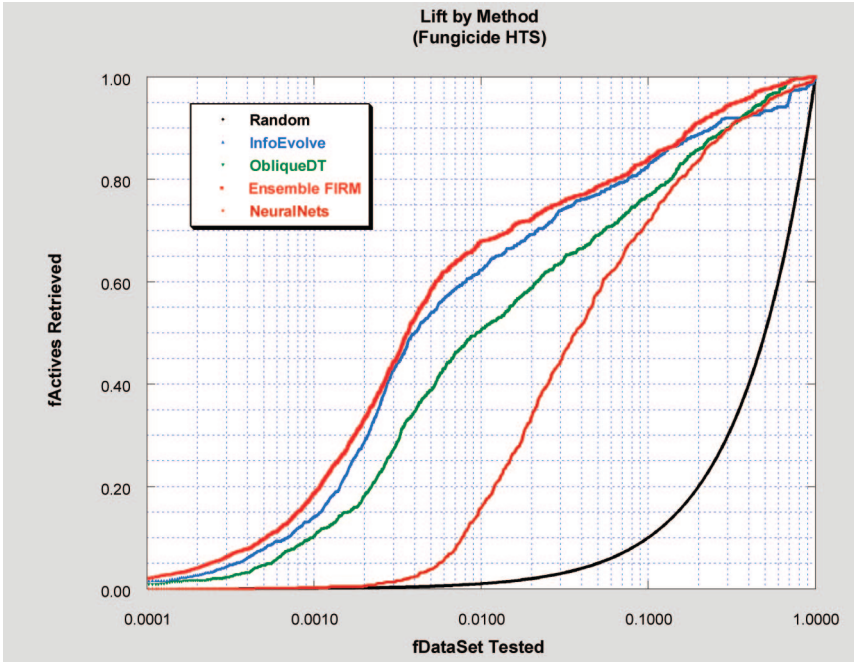

Figure 3. Lift charts for ensemble classifiers built using MolconnZ descriptors. The fraction of actives retrieved is plotted against the fraction of the data set tested on a log scale. Note that the "random" curve corresponds to retrieving x% of the actives when x% of the data set has been tested.

actual is minimized for the training data set, while monitoring a small validation set rms error in order to avoid overfitting. The single models are then combined into a single classifier by polling. The interested reader is referred to the work of Vaidyanathan[25] for additional discussion.

RESULTS

Each of the modeling techniques were used to train predictive models using at least one of the three WGB data sets. Once the trained models were deemed optimized by each method's advocate, the test data set was forecast using the optimal classifier. The classifier's prediction results were scored based upon the false positive error (true inactive forecast as active), the false negative error (true active forecast as inactive), the true positive rate (true active forecast as active), and the true negative rate (true inactive forecast as inactive) as well as the summed false error rates (false negative + false positive errors). These results are collected into Table 2. It is clear that all of the methods studied were capable of producing models that performed significantly better than random selection, on average demonstrating a True Positive rate of ca. 80% and a False Positive rate of ca. 13% when predicting the test data set. Modeling of data sets containing a higher proportion of inactive compounds leads to classifiers that are stronger on the majority class, as exemplified by the True Negative rates being highest for models trained on data set C (4:1 inactive/active) compared to data set B (2:1) or data set A (1:1) (e.g., FIRM, Oblique decision trees and C4.5). A direct comparison can be made between ensemble and single model methods by comparing the performance of Ensemble FIRM vs FIRM. For both of these methods the True Positive rate was comparable being ca. 81%, but there were significant improvements in the False Positive rate (5.3% vs 11.6−25.4%) for models developed with Ensemble FIRM.

The best results (minimum SumError) from any of the three data sets for each classification method are presented in Table 3 and Figure 2, sorted on the sum of the False Positive and False Negative errors. The methods achieving the best performance were those which created the final classifier from a family of models (Ensemble FIRM and InfoEvolve). There is little performance difference among the methods using single models to build the classifier. kNN analysis, often the basis of diversity and similarity analyses within the cheminformatics community,[30,31] was the third best method of all studied.

In the next phase of the study, we selected FIRM, Ensemble FIRM, InfoEvolve, Neural Networks, and the Oblique Classifier methods for further study by building models from the entire HTS WGB data set (629K compounds) for both atom-pair and MolconnZ[32] descriptors. The WGB HTS data set was randomly split into training (424K - 67%) and test (205K - 33%) subsets. Classifiers were developed using the training subset, and their performance was evaluated using the test subset. Ensemble-based classifiers were developed using all four methods for the MolconnZ descriptor set using the resampling technique discussed earlier. The overall accuracy was assessed using FalseNeg, FalsePos, and SumError measures, and the results are gathered into Table 4.

Neural networks can experience problems converging with wide data sets (i.e., data sets with a large number of descriptors relative to the number of data points), and our implementation is no exception. We could not successfully develop neural network models using the atom-pair descriptors (825 inputs) but could with the MolconnZ descriptors (221 inputs). We observed the same general performance ranking for these four methods for the HTS data set as had been seen in the earlier samplings from the HTS data set. As before ensemble-based classifiers performed better than single-model methods.

The test set predictions from each of the MolconnZ ensemble classifiers were converted into lift charts by sorting the test data sets descending on the prediction probabilities

and then counting the number of true actives as one traverses the ordered list from the topmost rated candidates to the worst (Figure 3). The expected performance of a random selection has been included for reference.

Inspection reveals that 80% of the active compounds in the test data set were delivered at efficiencies significantly greater than expected from random selection. For example, only the top 6% of the Ensemble FIRM ranked compounds need to be retrieved in order to obtain over 80% of the active compounds. This corresponds to an enrichment factor of thirteen. These results were quite encouraging, and classifiers trained on a more extensive set of HTS results and chemical descriptors were developed. Our findings and experiences developing and applying these classifiers in compound acquisitions will be the subject of a further communication.

CONCLUSIONS

We have compared a number of machine learning and chemometric tools for their efficiency and effectiveness at extracting useful information from a difficult but representative data set from agrochemical high throughput screening. Within the methods studied significant differences in performance are observed when models are applied to forecast an out of sample test set. Ensemble based methods were especially effective at achieving low False Positive rates, a significant outcome if such models are to be useful at focusing screening efforts of large potential chemistry collections. All of the methods were decidedly superior to random sampling, and ensemble-based classifiers constructed from several traditional machine learning methods have demonstrated the ability to achieve True Positive rates in excess of 80% while maintaining False Positive error rates in the 5−7% range.

REFERENCES AND NOTES

(1) Luco, J. M. Prediction of the Brain-Blood Distribution of a Large Set of Drugs from Structurally Derived Descriptors Using Partial Least-Squares (PLS) Modeling. *J. Chem. Inf. Comput. Sci.* **1999**, *39* (2), 396–404.

(2) Basak, S. C.; Grunwald, G. D.; Gute, B. D.; Balasubramanian, K.; Opitz, D. Use of Statistical and Neural Net Approaches in Predicting Toxicity of Chemicals. *J. Chem. Inf. Comput. Sci.* **2000**, *40* (4), 885–890.

(3) English, N. J.; Carroll, D. G. Prediction of Henry's Law Constants by a Quantitative Structure Property Relationship and Neural Networks. *J. Chem. Inf. Comput. Sci.* **2001**, *41* (5), 1150–1161.

(4) Mosier, P. D.; Jurs, P. C. QSAR/QSPR Studies Using Probabilistic Neural Networks and Generalized Regression Neural Networks. *J. Chem. Inf. Comput. Sci.* **2002**, *42* (6), 1460–1470.

(5) Lind, P.; Maltseva, T. Support Vector Machines for the Estimation of Aqueous Solubility. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (6), 1855–1859.

(6) Sutherland, J. J.; O'Brien, L. A.; Weaver, D. F. Spline-Fitting with a Genetic Algorithm: A Method for Developing Classification Structure-Activity Relationships. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (6), 1906–1915.

(7) Beger, R. D.; Young, J. F.; Fang, H. Discriminant Function Analyses of Liver-Specific Carcinogens. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (3), 1107–1110.

(8) Zheng, W.; Tropsha, A. Novel Variable Selection Quantitative Structure-Property Relationship Approach Based on the k-Nearest-Neighbor Principle. *J. Chem. Inf. Comput. Sci.* **2000**, *40* (1), 185–194.

(9) Simmons, K. Empirical Validation of the Effectiveness of Chemical Descriptors in Data Mining, 2nd Joint Sheffield Conference on Cheminformatics: Computational Tools for Lead Discovery, University of Sheffield, Sheffield, U.K., April 9, 2001.

(10) Bruce, C. L.; Melville, J. L.; Pickett, S. D.; Hirst, J. D. Contemporary QSAR Classifiers Compared. *J. Chem. Inf. Model.* **2007**, *47* (1), 219–227.

(11) Plewczynski, D.; Spieser, S. A. H.; Koch, U. Assessing Different Classification Methods for Virtual Screening. *J. Chem. Inf. Model.* **2006**, *46* (3), 1098–1106.

(12) Bender, A.; Glen, R. C. A Discussion of Measures of Enrichment in Virtual Screening: Comparing the Information Content of Descriptors with Increasing Levels of Sophistication. *J. Chem. Inf. Model.* **2005**, *45* (5), 1369–1375.

(13) Yao, X. J.; Panaye, A.; Doucet, J. P.; Zhang, R. S.; Chen, H. F.; Liu, M. C.; Hu, Z. D.; Fan, B. T. Comparative Study of QSAR/QSPR Correlations Using Support Vector Machines, Radial Basis Function Neural Networks, and Multiple Linear Regression. *J. Chem. Inf. Comput. Sci.* **2004**, *44* (4), 1257–1266.

(14) Feng, J.; Lurati, L.; Ouyang, H.; Robinson, T.; Wang, Y.; Yuan, S.; Young, S. Predictive Toxicology: Benchmarking Molecular Descriptors and Statistical Methods. *J. Chem. Inf. Comput. Sci.* **2003**, *43* (5), 1463–1470.

(15) Simmons, K. Practical Outcomes of Data Mining in-Vivo HTS Data, Proceedings of the Conference on Exploiting Molecular Diversity, San Diego, CA, Cambridge Health Tech, 2002.

(16) Pipeline Pilot, version 4, Scitegic. http://www.scitegic.com (accessed July 19, 2008).

(17) MDL connection table specifications available at MDL, Inc. http://www.mdli.com (accessed July 19, 2008).

(18) Concord is available from Tripos, Inc. http://www.tripos.com/data/SYBYL/Concord_072505.pdf (accessed July 19, 2008). For the specifications of the mol2 format, see: http://tripos.com/tripos_resources/fileroot/mol2_format_Dec07.pdf (accessed July 19, 2008).

(19) Todeschini, R.; Consonni, V. *Handbook of Molecular Descriptors*; Mannhold, R., Kubinyi, H., Timmerman, H., Eds.; WILEY-VCH Verlag Gmbh:Weinheim, Germany, 2000.

(20) Carhart, R.; Smith, D. H.; Venkataraghavan, R. J. Atom pairs as molecular features in structure-activity studies: definition and applications. *J. Chem. Inf. Comput. Sci.* **1985**, *25*, 64–73.

(21) Many thanks to Prof. Alex Tropsha, University of North Carolina.

(22) BASE SAS, Version 6; SAS Institute Inc.: Cary, NC, http://www.sas.com (accessed July 19, 2008).

(23) PLS Toolbox for MatLab; Eigenvector Research, Inc.: 3905 West Eaglerock Drive, Wenatchee, WA 98801. http://www.eigenvector.com (accessed July 19, 2008).

(24) Owens, A. J.; Filkin, D. L. Efficient training of the Back Propagation Network by solving a system of stiff ordinary differential equations. International Joint Conference on Neural Networks, II, Washington, DC, 1989; pp 381−386.

(25) Vaidyanathan, G. InfoEvolve - Moving from Data to Knowledge Using Information Theory and Genetic Algorithms. *Ann. N.Y. Acad. Sci.* **2004**, *1020*, 227–238.

(26) Hawkins, D. Formal Inference-Based Recursive Modeling, version 2.3; Univ. of Minnesota: Duluth, MN, 1999.

(27) Murphy, S. K.; Kasif, K.; Salzberg, S. A System for Induction of Oblique Decision Trees. *J. Artificial Intelligence Res.* **1994**, *2*, 1–32.

(28) Quinlan, J. R. *C4.5: Programs for Machine Learning*; Morgan Kaufmann:San Mateo, CA, 1993.

(29) CART Pro 6; Salford Systems: San Diego, CA. 2006.http://www.salford-systems.com (accessed July 19, 2008).

(30) Brown, R.; Martin, Y. Use of Structure-Activity Data to Compare Structure-Based Clustering Methods and Descriptors for Use in Compound Selection. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 572–584.

(31) Brown, R.; Martin, Y. The Information Content of 2D and 3D Structural Descriptors Relevant to Ligand-Receptor Binding. *J. Chem. Inf. Comput. Sci.* **1997**, *37*, 1–9.

(32) MolconnZ version 3.50; EduSoft: Ashland, VA. http://www.edusoft-lc.com/molconn (accessed July 19, 2008).

CI800142D