

Accurate and Interpretable Computational Modeling of Chemical Mutagenicity

James J. Langham* and Ajay N. Jain

Cancer Research Institute, University of California, San Francisco, 2340 Sutter Street, San Francisco, California 94143-0128

Received March 14, 2008

We describe a method for modeling chemical mutagenicity in terms of simple rules based on molecular features. A classification model was built using a rule-based ensemble method called RuleFit, developed by Friedman and Popescu. We show how performance compares favorably against literature methods. Performance was measured through the use of cross-validation and testing on external test sets. All data sets used are publicly available. The method automatically generated transparent rules in terms of molecular structure that agree well with known toxicology. While we have focused on chemical mutagenicity in demonstrating this method, we anticipate that it may be more generally useful in modeling other molecular properties such as other types of chemical toxicity.

INTRODUCTION

The ability to accurately estimate a molecule's properties given molecular structure has been a goal for over a century.¹ Methods to estimate molecular properties continue to have great relevance. For example, being able to quickly estimate toxicity problems for drug candidates could be very useful because unforeseen toxicity is a major cause of failure for late stage drug candidates.² Methods that can also help to elucidate molecular features associated with a property may prove useful for designing and optimizing candidate molecules. If it is known that a particular molecular feature in a candidate molecule is associated with toxicity, then one could substitute it with an appropriate nontoxic feature.

In this paper we present a method for modeling chemical mutagenicity, the ability of a chemical to cause genetic mutations, and show how the models may be interpreted in terms of simple molecular features familiar to chemists. For model building and testing we used two publicly available mutagenicity data sets. Results are compared with those of two other recent studies using the same data sets. In spite of the simplicity of the chemical descriptor used we achieved accurate results as compared with the other studies. Also we show our methods to be fast and therefore suitable for screening large data sets.

In addition we used our method to classify a set of approved drugs and a screening library, the ZINC druglike subset.²⁴ We screened these data sets for two reasons. First, to quantify the extent to which our method might, presumably incorrectly, identify approved drugs as mutagens. Second, to explore the utility of the method for prescreening databases of molecules used for virtual screening lead discovery exercises in order to remove likely mutagens. These results suggest that the method may be a useful filter to identify a conservative subset of compounds from screening databases that are significantly less likely to harbor mutagens.

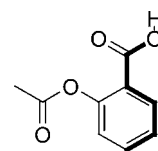


Figure 1. An example of the atom pair feature O3_1_D5_C2_Ar2 in the aspirin structural diagram.

METHODS

Our approach was to use a simple molecular descriptor in combination with several classification methods. The molecular descriptor used is a type of “Atom Pairs” descriptor.³ The classification methods used were the RuleFit,⁴ support vector machine (SVM),⁵ and K-nearest neighbor (KNN) algorithms.

Molecular Descriptor. Given a set of molecules molecular descriptors were generated by enumerating all combinations of pairs of atoms and finding the shortest path length along bonds between each pair. For a set of molecules a set of descriptors are generated in the following form:

<atom 1 description> - <distance> - <atom 2 description>

This is the same definition as used by Carhart, Smith, and Venkataraghavan in their “Atom Pairs” method.³ Our implementation differs slightly from theirs in the way the atoms are described. We describe each atom by the atom's element, hybridization, aromaticity, and number of heavy atom neighbors. An example is given in Figure 1 in which a particular atom pair feature is marked in the structural diagram of aspirin. This feature is O3_1_D5_C2_Ar2 which describes a sp³ hybridized oxygen atom with one heavy atom neighbor separated by five bond lengths from an aromatic carbon atom with two heavy atom neighbors. The atom description part of the feature lists the element symbol, hybridization (codes 1, 2, or 3 for sp, sp², and sp³, respectively), whether the atom is aromatic (“Ar” if aromatic), and number of heavy atom neighbors. Each molecule is then described by a feature vector with each component in the vector being either 1 if the atom pair feature is present

* Corresponding author e-mail: james.langham@jainlab.org.

in the molecule or 0 if it is not. The resulting features are then given as input to the classification methods. Features were generated using computer software written in C++ linking to the Open Babel chemistry toolkit library (version 2.1.1).^{6,7} Software is available upon request.

We chose to use this particular descriptor because of its ease of interpretability and the success of similar descriptors in modeling molecular properties.^{3,8,9} Since the features are derived directly from a molecule's structural diagram, they encode concepts that are familiar to chemists such as the type of hybridization and whether an atom is primary, secondary, or tertiary. Another useful feature of this descriptor is that it would be possible to easily extend it to larger substructures than atom pairs.

Classification Algorithms. The main classification method used was the RuleFit algorithm, a relatively recent method, chosen because it appeared to offer both good accuracy and good model interpretability. This method is a type of ensemble learning method in which effective classifiers are constructed from ensembles of simpler classifiers.¹⁰ The details of the RuleFit method are described in the papers by Friedman and Popescu.^{4,11} We give a brief overview of the method here in order to explain how we used it in this study.

The RuleFit method builds classifiers as ensembles of rules. Each rule takes the form of a conjunction of simple tests of the value of certain input variables. In the case of this study each rule would be a test of whether a particular feature was present in a molecule to be classified (test molecule). The rule returns a value of either 1 for the case in which all the tests in a rule are true and 0 otherwise. This may be expressed mathematically by eq 1 which describes a rule, r , as a function of the feature vector \mathbf{x} and with $r(\mathbf{x}) \in \{0, 1\}$. The term I is an indicator function that tests whether a particular feature x_j is present or not in the test molecule, and n is the number of features in the rule.

$$r(\mathbf{x}) = \prod_{j=1}^n I(x_j \in \{0, 1\}) \quad (1)$$

An example of a possible molecular rule is given in eq 2.

$$r(\mathbf{x}) = \begin{cases} I(\text{A nitro group is present}) \\ \cdot I(\text{A phenyl ring is not present}) \\ \cdot I(\text{A carbonyl group is present}) \end{cases} \quad (2)$$

Classification of a test molecule is achieved using eq 3 for an ensemble of K rules. The terms $\{\hat{a}_k\}_0^K$ are parameters that are determined during the training of the classifier. The process of building a RuleFit classifier is accomplished in two major steps. The first step is to generate an ensemble of decision trees of randomly determined sizes. The rules r are inferred from the decision trees as paths within the decision trees.⁴ Once the decision trees have been generated, the rules are specified. The next major step is to fit coefficients to the rules, that is find the parameters in eq 3. The details of the method are described in the papers of Friedman and Popescu.^{4,11}

$$F(x) = \hat{a}_0 + \sum_{k=1}^K \hat{a}_k r_k(x) \quad (3)$$

We used the implementation of the RuleFit method written by the authors of the method which consists of an executable binary designed to be used in combination with the R

statistical package.¹² (An open source implementation of RuleFit, written by other programmers, is also available as part of the TMVA statistical package¹³). Version R/RuleFit beta (8/10/05) with R version 2.5.1 was used.

In addition to RuleFit we also used two other classification methods with the same descriptors for comparison. These methods were support vector machine (SVM) classification⁵ and K-nearest neighbor (KNN) classification. The SVM implementation used was the LIBSVM library (version 2.85).¹⁴ We used a linear kernel to build models. The KNN implementation used was from the "class" package for R.¹⁵ We used the ROCR package for R to calculate performance measures such as ROC plots.¹⁶ An external method, LAZAR,¹⁷ was also used for further comparison. LAZAR uses linear substructures as the molecular descriptor and a form of nearest neighbor classification to perform classification.

Data Sets. We used two publicly available mutagenicity data sets. One was used for training and performing initial validation of mutagenicity models. The second was used for doing external testing as the last step in measuring method accuracy.

The training data set had been prepared by Bursi and co-workers¹⁸ and contained 4337 diverse organic compounds, 2401 mutagens and 1936 nonmutagens. Bursi and co-workers had used this data set to identify substructures (called toxicophores) that could be used to help classify whether test compounds were mutagenic. In their study they identified toxicophores from this set of 4337 compounds which they then used on an external data set to validate their method. This external data set was not publicly available at the time of our study.

The external data set we used was based on the Carcinogenic Potency Database (CPDB).¹⁹ The actual version we used was an annotated version from the USEPA's DSSTox^{20,21} database network, specifically the CPDBAS (version 5a) data set. This file was prepared by only including organic compounds and removing any compounds with the same calculated InChi identifier²² as any of the compounds from the training set. This resulted in a set of 400 compounds, with 174 mutagens and 226 nonmutagens. This filtering was carried out using C++ software linked to the Open Babel library (version 2.1.1).⁷

The data set of small-molecule drugs (drug data set) was a set of 962 approved small-molecule drugs.²³ We also tested against the druglike subset of the ZINC virtual screening library (ZINC data set).²⁴ We randomly selected 1% of the ZINC druglike subset of 2 million compounds for testing.

EXPERIMENTAL SECTION

We trained classifiers using the training data set and did initial validation with 2-fold cross-validation. The performance metrics used were receiver operator characteristic (ROC) area under the curve (AUC) and accuracy.²⁵ Accuracy is the fraction of classifier predictions that were correct. ROC AUC is a useful metric for evaluating classifiers and is an estimate of the probability that the classifier ranks randomly chosen positive examples higher than randomly chosen negative examples.²⁶ A value of 1.00 for a classifier indicates optimal performance, while 0.50 indicates the classifier's performance is no better than a random method. We "tuned" classifier parameters to try and improve AUC performance

Table 1. Test Statistics Resulting from 2-Fold Cross-Validation for Classifiers on the Training Data Set^a

method	AUC	AUC C.I.	accuracy
KNN ($k = 1$)	0.780	0.766 – 0.793	0.771
KNN ($k = 5$)	0.829	0.816 – 0.841	0.756
KNN ($k = 10$)	0.813	0.800 – 0.825	0.733
KNN ($k = 15$)	0.812	0.799 – 0.824	0.740
RuleFit (default)	0.866	0.855 – 0.877	0.791
RuleFit (tuned)	0.865	0.854 – 0.876	0.792
SVM-linear (default)	0.842	0.831 – 0.854	0.777
SVM-linear (tuned)	0.840	0.828 – 0.851	0.774

^a Confidence intervals (95%) for the AUC values are shown under the heading C.I.

by tuning on the individual folds. In the case of the linear kernel SVM classifier we adjusted the parameter C, which weights the effects of misclassifications, to maximize the AUC value. For RuleFit we adjusted the “path.inc” parameter which controls the step size in the gradient directed search for model parameters. For RuleFit training we selected the classification mode and indicated that all input variables were categorical variables.

After concluding the initial validation and tuning we trained the RuleFit and SVM classifiers and tested on the external CPDB data set. We used the LAZAR method¹⁷ to classify the external set. The RuleFit and SVM classifiers were also used to score the drug data set and ZINC data set.

RESULTS

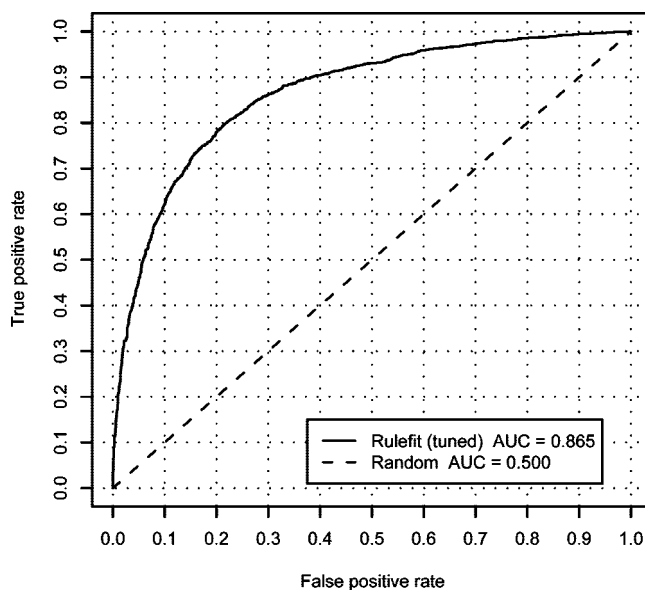
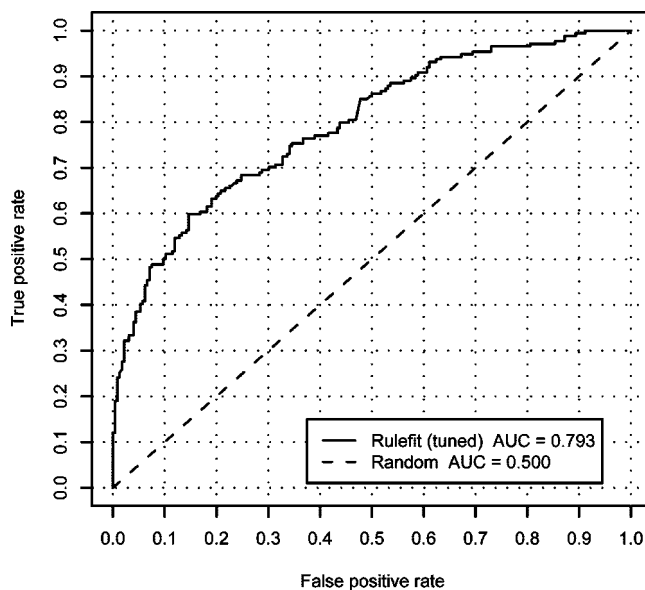
Classification Performance. The results from the initial training and validation of the training data set are shown in Table 1. A total of 9634 distinct atom pair features were generated on this data set. Results for SVM and RuleFit are shown both before and after the parameter tuning. Confidence intervals (95%) were estimated using resampling. The attempt to improve the RuleFit classifier performance through tuning resulted in no improvement. The AUC value went from 0.866 before tuning to 0.865 after tuning. The SVM classifier performance went from 0.842 before tuning to 0.868 after tuning. Both RuleFit and SVM classification performed better than KNN classification.

The ROC curves based on the tuned RuleFit classifier are shown in Figures 2 and 3. The curve shows good enrichment for high classifier scores, rising rapidly near the origin of the plot.

Bursi and co-workers used the same data set to find a set of substructures that could be used to predict mutagenicity. After finding the set of substructures they were able to classify the same data set they had used to find substructures with an accuracy of 0.81. The cross-validation accuracy obtained with our method is roughly similar with an accuracy of 0.792 obtained for the RuleFit (tuned) method.

Three models were used to classify the external data set. These were the tuned RuleFit model and the tuned and untuned linear SVM models. All three models were retrained using the entire training data set at this stage.

Classifying the External Data Sets. The results from testing the classifiers on the external CPDB data set are shown in Table 2. LAZAR used 194,293 features for its model. The tuned linear SVM classifier performed the best with respect to the ROC AUC value with a value of 0.839.

**Figure 2.** ROC curve for the 2-fold cross-validation results on the training data set using RuleFit (tuned).**Figure 3.** ROC plot of the RuleFit (tuned) classifier on the external CPDB data set.**Table 2.** Performance of the Classifiers on the External CPDB Data Set^a

method	AUC	AUC C.I.	accuracy
RuleFit (tuned)	0.793	0.743–0.833	0.718
SVM-linear (untuned)	0.825	0.783–0.864	0.770
SVM-linear (tuned)	0.839	0.780–0.875	0.745
LAZAR	0.806	0.760–0.846	0.722

^a Confidence intervals (95%) for the AUC values are shown under the heading C.I.

Both the RuleFit method and LAZAR performed slightly worse with ROC AUCs of 0.793 and 0.806, respectively. However the confidence intervals indicate no statistically significant difference between the methods on this data set.

Interpretation of Classification Models. The RuleFit model that was built using all the training data used only 228 of the original 9634 input variables. The model contained 308 rules. During the training of a RuleFit model the majority

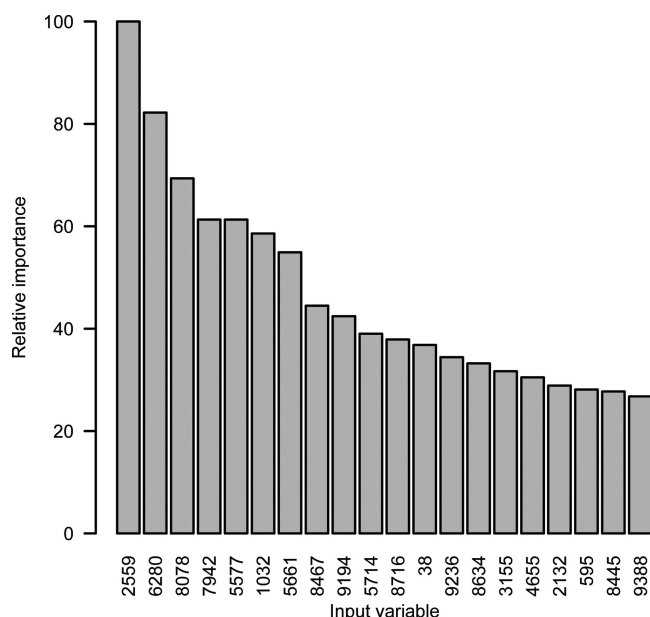


Figure 4. Relative importance of the top 20 input variables in the RuleFit (tuned) mutagenicity model.

of the input variables and initial rules are typically pruned from the model.⁴ That is most of the coefficients $\{\hat{a}_k\}_1^K$ from eq 3 are set to zero. The magnitude of the remaining nonzero coefficients gives the relative importance of the rule in the model, and the sign of the coefficients indicates whether the rule is associated with mutagenicity or nonmutagenicity. The importance I_k of a particular rule r_k is given by eq 4. The term s_k is the rule support, the fraction of training examples in which the rule r_k holds. The support is defined by eq 5 in which N is the number of training examples.

$$I_k = |\hat{a}_k| \cdot \sqrt{s_k(1 - s_k)} \quad (4)$$

$$s_k = \frac{1}{N} \sum_{i=1}^N r_k(\mathbf{x}_i) \quad (5)$$

The importance J_l of a particular input variable x_l in the model can be estimated using eq 6. This equation sums the importances of the rules in which the input variable occurs. The term m_k is the number of input variables present in a rule and serves to normalize the importance of input variables over rules with differing numbers of input variables.

$$J_l = \sum_{x_l \in r_k} I_k / m_k \quad (6)$$

The fact that only 228 of the original 9634 input variables were retained in the final model makes the job of interpretation easier, having fewer variables to examine. Figure 4 shows the 20 highest ranked variables with importance being measured by eq 6. Table 3 shows the features associated with these top 20 input variables, and Figure 5 shows examples of the top five input variables mapped onto the mutagen 1-nitro-8-nitrosopyrene.

Some of these features are associated with mutagens and some with nonmutagens. Examination of the rules in which they appear gives more information about their role in classification. Examination of Table 3 reveals that many of these features are related to the known toxicology of mutagenicity. For example, eight of these features describe pairs of aromatic carbon atoms (for example Features 2559

Table 3. Top 20 Ranked Input Variables from the RuleFit (Tuned) Mutagenicity Model

rank	ID	feature
1	2559	C2_Ar2_D4_C2_Ar2
2	6280	O2_1_D1_N2_3
3	8078	N2_3_D1_C2_Ar3
4	7942	C2_Ar3_D4_C2_Ar2
5	5577	O2_1_D1_N2_2
6	1032	O3_2_D1_C3_3
7	5661	C2_Ar2_D5_C2_Ar2
8	8467	C2_Ar3_D5_C2_Ar3
9	9194	C2_Ar3_D5_C2_Ar2
10	5714	C3_1_D2_C3_1
11	8716	ClO_1_D1_C2_Ar3
12	38	ClO_1_D2_C3_2
13	9236	C2_Ar3_D4_C2_Ar3
14	8634	C2_Ar3_D4_C2_Ar3
15	3155	ClO_1_D1_C2_3
16	4655	O2_1_D2_O2_1
17	2132	O2_1_D1_C2_3
18	595	O2_1_D2_C2_Ar2
19	8445	S3_4_D1_C2_Ar3
20	9388	C2_Ar2_D8_C2_Ar2

and 7942). This indicates the presence of polycyclic aromatic systems which often act as mutagens because of their ability to intercalate into DNA.¹⁸ Features 6280 and 5577 indicate a nitro and nitroso group, respectively. Both features are highly associated with mutagenicity when connected to an aromatic system. Bursi and co-workers cite a possible mechanism for the mutagenicity of these features, their ability to form DNA-binding electrophilic intermediates.¹⁸

We also examined several of the highest weighted input variables in combination with the rules in which they occurred. Figure 6 shows two mutagens with selected features highlighted. These features appear in model rules that associate the presence of these features with mutagenicity. Bursi and co-workers identified these three substructures as toxicophores that could be used to help predict mutagenicity.¹⁸

Figure 7 shows two nonmutagens with selected features highlighted. These features appeared in rules that associate their presence with nonmutagenicity. These features were identified by Bursi and co-workers as “detoxifying substructures” with respect to mutagenicity, probably due to electron-withdrawing or steric effects.¹⁸

A SVM model with a linear kernel can also be analyzed by examining the weights on the input variables. We found that many of the same variables that were highly weighted in the RuleFit model were also highly weighted in the tuned SVM model. In the SVM model however there were many more input variables with nonzero weights making the analysis more difficult.

Testing on Approved Drugs and the ZINC Library.

We used the statistical models to classify approved drugs in order to test whether our approach might be applicable to druglike molecules. We had Ames testing data (from the training and external test sets) for 110 of the 962 approved drugs that we tested the models against. Of these, 30 had been tested to be positive and 80 negative by the Ames test. Since we only knew the “true” labels for 11% of the data set, we could not definitively measure the overall accuracy of the predictions. The RuleFit model predicted 22% of the drug data set to be mutagens, and the tuned SVM model

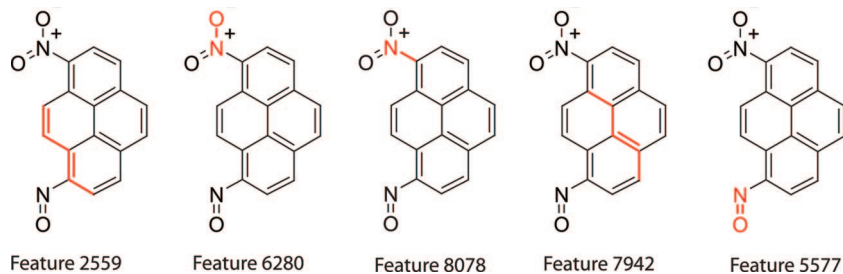


Figure 5. The top 5 ranked features are shown in red on the mutagen 1-nitro-8-nitrosopyrene. Some of the features occur more than once in the molecule, but only one example of each feature is shown.

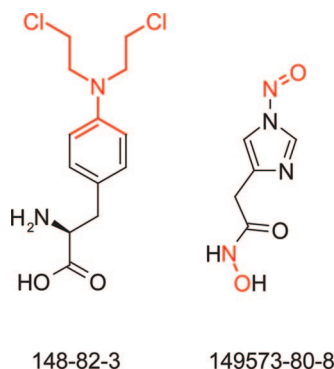


Figure 6. Two mutagens are shown with mutagenic features highlighted in red. 148-82-3 contains the feature Cl0_1_D5_C2_Ar2, and 149573-80-8 contains the features O3_1_D1_N2_2 and O2_1_D1_N2_2.

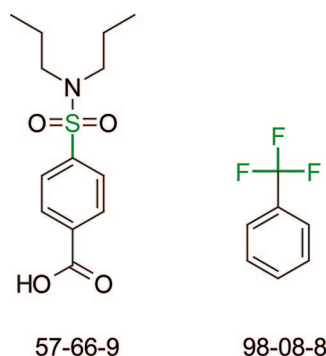


Figure 7. Two nonmutagens are shown with nonmutagenic features highlighted in green. 57-66-9 contains the feature S3_4_D1_C2_Ar3, while 98-08-8 contains the feature F0_1_D2_F0_1.

predicted 21% to be mutagens. Many of the highest scoring molecules were antineoplastic agents and suspected carcinogens. For example the three highest-scoring drugs were carmustine, melphalan, and lomustine which are all antineoplastic agents and known or suspected carcinogens. It should be noted that of these three drugs only melphalan was not present in the training set. Melphalan has been described as a potential carcinogen in the DrugBank annotation.^{27,28} Another high-scoring drug not present in the training set was the anesthetic phenazopyridine which is also described as a potential carcinogen by DrugBank. These two examples show that the method was able to make correct predictions for previously unseen pharmaceuticals.

The RuleFit model was applied to the ZINC data set and estimated 26.4% of the molecules to be mutagens. The cumulative distributions of RuleFit scores for the ZINC data set, the drug data set, and known mutagen and nonmutagen drugs are shown in Figure 8.

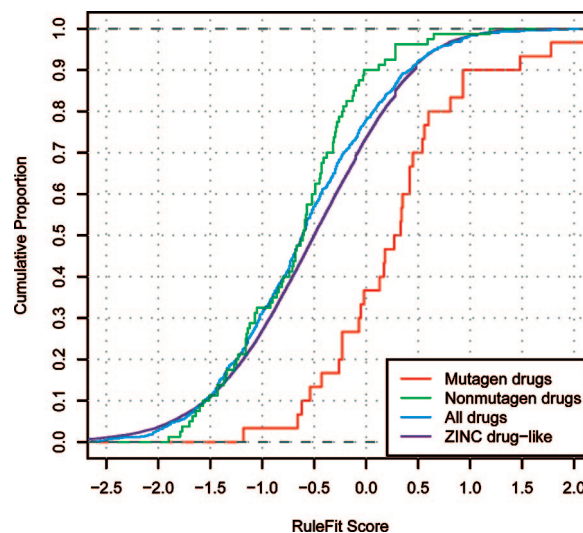


Figure 8. Cumulative distributions of Rulefit scores for the ZINC data set, drug data, mutagen drugs, and nonmutagen drugs.

Training and Testing Speed. For the following times a Linux workstation with a 2.0 GHz Intel dual core Xeon processor was used. It took 6 CPU ms per molecule, on average, to discover the features over the training data set. This time includes the time it took to read the structures from a file. Training the RuleFit classifier on the training data set of 4337 molecules (9634 features) took 350 CPU s. Testing the resulting RuleFit classifier on the training data set took 1.9 CPU s or 0.4 CPU ms per molecule. Training the SVM classifier on the training data took 32 CPU ms, and testing took 12 CPU ms or 3 CPU μ s per molecule.

DISCUSSION

We have shown that the combination of a simple molecular descriptor combined with effective statistical methods was able to classify molecules for mutagenicity quickly and accurately. Performance on an external test set was competitive with the LAZAR method, a recent mutagenicity method described in the literature. In addition to accuracy it was shown to be possible to interpret the highly ranked features of the model in terms of simple molecular features that agreed with known toxicology.

While performance was good, it was clearly not optimal. The best accuracy value was only 0.770. One possible reason for this is differences in the way the Ames method is executed in different laboratories. Both the Bursi and CPDB data sets contain data from multiple laboratories, and interlaboratory Ames test error has been estimated to be about 15%¹⁸ corresponding to a maximal possible accuracy of 0.850.

Therefore it is likely that the best practical performance on these data sets is well below the theoretical maximum. In addition better performance may have been obtained with a more complex descriptor. We have shown that the descriptor used was able to describe many of the important molecular features involved in mutagenicity such as nitro and nitroso groups. However some other important mutagenic features, such as three-membered epoxide and aziridine rings, would be missed by the two vertex descriptor used. To address this issue we experimented with using larger substructures than atom pairs. Our preliminary work in this area did not find an increase in accuracy. For example we tried using the set of connected "atom triples" in effect adding an edge to the atom pair substructure. This resulted in the number of distinct features derived from the training data set going from 9634 to 1,836,781. Performance, as measured by 2-fold cross-validation, did not improve. In spite of this result, it seems quite possible that a method to selectively pick out discriminative substructures in an efficient way would lead to improved results.

The results on approved drugs showed that the method predicted mutagenicity in several compounds not present in the training set which were also suspected carcinogens. We were surprised that such a large percentage of approved drugs (21–22%) was predicted to be mutagens. Our expectation had been that a relatively small number of known drugs would be classified as mutagens. Several of the predictions were confirmed to be either mutagens or possible carcinogens as described above. Also, for the 110 drugs (of 962 total) for which an Ames test result was available, 30 (27%) had positive Ames test results. This shows that at least 3% of the drug data set are mutagens and suggests that a significant number of approved drugs may give a positive Ames test result.

Figure 8 indicates that the Rulefit score may be useful for the purpose of filtering likely mutagens from a screening library such as ZINC. This figure shows there is a clear difference between the scores of the mutagen and nonmutagen drugs. If a Rulefit score of zero was used as a threshold for filtering, then the figure suggests that 90% of nonmutagens would be retained while 70% of mutagens would be discarded.

In using this method for virtual screening for mutagenicity it would probably be most practical to treat the highest scoring predictions as the most reliable. Consideration of the specific molecular features resulting in a high model score may help to corroborate the prediction against known chemistry. It may also be possible to exploit such knowledge to make modifications to candidate molecules in order to optimize their properties. A very simple example of this is suggested by Figure 7 in which the addition of a trifluoro group to mutagenic benzene results in a nonmutagen.

CONCLUSION

Overall the method presented gave accurate and fast predictions of chemical mutagenicity for a diverse set of molecules. The RuleFit method was very useful in simplifying the interpretation of the resulting models and helping to automatically identify molecular features important for mutagenicity. We believe that this statistical method has potential as a useful tool for building models of molecular

properties and helping to uncover the relationship between molecular features and specific properties.

ACKNOWLEDGMENT

The authors gratefully acknowledge the NIH for financial support (grant GM070481). The authors wish to thank Prof. Jerome Friedman for making the RuleFit software available and Christoph Helma for making the LAZAR software available.

REFERENCES AND NOTES

- (1) Crum Brown, A.; Fraser, T. On the Connection Between Chemical Constitution and Physiologic Action. Part 1. On the Physiological Action of Salts of the Ammonium Bases, Derived from Strychnia, Brucia, Thebia, Codeia, Morphia and Nicotia. *Trans. R. Soc. Edinburgh* **1868**, 25, 151–203.
- (2) Kramer, J. A.; Sagartz, J. E.; Morris, D. L. The Application of Discovery Toxicology and Pathology Towards the Design of Safer Pharmaceutical Lead Candidates. *Nat. Rev. Drug Discovery* **2007**, 6, 636–649.
- (3) Carhart, R. E.; Smith, D. H.; Venkataraghavan, R. Atom Pairs as Molecular Features in Structure-Activity Studies: Definition and Applications. *J. Chem. Inf. Comput. Sci.* **1985**, 25, 64–73.
- (4) Friedman, J. H.; Popescu, B. E. Predictive Learning via Rule Ensembles; Technical Report; Department of Statistics, Stanford University: 2005.
- (5) Cortes, C.; Vapnik, V. Support-Vector Networks. *Machine Learning* **1995**, 20, 273–297.
- (6) Guha, R.; Howard, M. T.; Hutchison, G. R.; Murray-Rust, P.; Rzepa, H.; Steinbeck, C.; Wegner, J.; Willighagen, E. L. The Blue Obelisk-Interoperability in Chemical Informatics. *J. Chem. Inf. Model.* **2006**, 46, 991–998.
- (7) *Open Babel Package, version 2.1.1*; 2007.
- (8) Aronov, A. M.; Goldman, B. B. A Model for Identifying HERG K⁺ channel Blockers. *Bioorg. Med. Chem.* **2004**, 12, 2307–2315.
- (9) Langham, J. J. Discovering Drug Candidates in Virtual Chemical Libraries: A Novel Graph-Based Method for Virtual Screening, Ph.D. Thesis, University of California Santa Cruz: CA, 2006.
- (10) Hastie, T.; Tibshirani, R.; Friedman, J. Boosting and Additive Trees. In *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*; Springer, 2001; pp 299–346.
- (11) Friedman, J. H.; Popescu, B. E. Gradient Directed Regularization for Linear Regression and Classification; Technical Report; Department of Statistics, Stanford University: 2004.
- (12) R Development Core Team. *R: A Language and Environment for Statistical Computing version 2.5.1*; R Foundation for Statistical Computing: Vienna, Austria, 2007.
- (13) Höcker, A.; Speckmayer, P.; Stelzer, J.; Tegenfeldt, F.; Voss H.; Voss, K. *TMVA Toolkit for Multivariate Data Analysis with ROOT*; 2007, <http://tmva.sourceforge.net/>.
- (14) Chang, C.-C.; Lin, C.-J. *LIBSVM: A Library for Support Vector Machines version 2.85*; 2007, <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- (15) Venables, W.; Ripley, B. Classification. In *Modern Applied Statistics with S*, 4th ed.; Springer: 2002; pp 331–352.
- (16) Sing, T.; Sander, O.; Beerenwinkel, N.; Lengauer, T. ROC-R: Visualizing Classifier Performance in R. *Bioinformatics* **2005**, 21, 3940–3941.
- (17) Helma, C. Lazy Structure-Activity Relationships (Lazar) for the Prediction of Rodent Carcinogenicity and Salmonella Mutagenicity. *Mol. Diversity* **2006**, 10, 147–158.
- (18) Kazius, J.; McGuire, R.; Bursi, R. Derivation and Validation of Toxicophores for Mutagenicity Prediction. *J. Med. Chem.* **2005**, 48, 312–320.
- (19) Gold, L. S.; Slone, T. H.; Manley, N. B.; Garfinkel, G. B.; Hudes, E. S.; Rohrbach, L.; Ames, B. N. The Carcinogenic Potency Database: Analyses of 4000 Chronic Animal Cancer Experiments Published in the General Literature and by the U.S. National Cancer Institute/National Toxicology Program. *Environ. Health Perspect.* **1991**, 96, 11–15.
- (20) Richard, A. M.; Williams, C. R. Distributed Structure-Searchable Toxicity (DSSTox) Public Database Network: A Proposal. *Mutat. Res.* **2002**, 499, 27–52.
- (21) Richard, A. M.; Gold, L. S.; Nicklaus, M. C. Chemical Structure Indexing of Toxicity Data on the Internet: Moving Toward a Flat World. *Curr. Opin. Drug Discovery Devel.* **2006**, 9, 314–325.

- (22) Stein, S. E.; Heller, S. R.; Tchekhovskoi, D. Proceedings of the 2003 International Chemical Information Conference (Nimes), 2003; pp 131–143.
- (23) Cleves, A. E.; Jain, A. N. Robust Ligand-Based Modeling of the Biological Targets of Known Drugs. *J. Med. Chem.* **2006**, *49*, 2921–2938.
- (24) Irwin, J. J.; Shoichet, B. K. ZINC-A Free Database of Commercially Available Compounds for Virtual Screening. *J. Chem. Inf. Model* **2005**, *45*, 177–182 (accessed Feb 19, 2008).
- (25) Triballeau, N.; Acher, F.; Brabet, I.; Pin, J.-P.; Bertrand, H.-O. Virtual Screening Workflow Development Guided by the “Receiver Operating Characteristic” Curve Approach. Application to High-Throughput docking on Metabotropic Glutamate Receptor Subtype 4. *J. Med. Chem.* **2005**, *48*, 2534–2547.
- (26) Fawcett, T. ROC Graphs: Notes and Practical Considerations for Data Mining Researchers; Technical Report; Hewlett-Packard Company: 2003.
- (27) Wishart, D. S.; Knox, C.; Guo, A. C.; Shrivastava, S.; Hassanali, M.; Stothard, P.; Chang, Z.; Woolsey, J. DrugBank: A Comprehensive Resource for in Silico Drug Discovery and Exploration. *Nucleic Acids Res.* **2006**, *34*, D668–D672.
- (28) Wishart, D. S.; Knox, C.; Guo, A. C.; Cheng, D.; Shrivastava, S.; Tzur, D.; Gautam, B.; Hassanali, M. DrugBank: A Knowledgebase for Drugs, Drug Actions and Drug Targets. *Nucleic Acids Res.* **2008**, *36*, D901–D906.

CI800094A