IOPscience iopscience.iop.org

Home Search Collections Journals About Contact us My IOPscience

Machine-learning models for combinatorial catalyst discovery

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2005 Meas. Sci. Technol. 16 270

(http://iopscience.iop.org/0957-0233/16/1/035)

View the table of contents for this issue, or go to the journal homepage for more

Download details:

IP Address: 157.182.150.22
This content was downloaded on 26/06/2014 at 16:58

Please note that terms and conditions apply.

Meas. Sci. Technol. **16** (2005) 270–277 doi:10.1088/0957-0233/16/1/035

Machine-learning models for combinatorial catalyst discovery

Gregory A Landrum, Julie E Penzotti and Santosh Putta

Rational Discovery LLC, 555 Bryant St 467, Palo Alto, CA 94301, USA

Received 15 April 2004
Published 16 December 2004
Online at stacks.iop.org/MST/16/270

Abstract
A variety of machine learning algorithms, including hierarchical clustering, decision trees, *k*-nearest neighbours, support vector machines and bagging, were applied to construct models to predict the molecular weight of the polymers produced by a set of 96 homogeneous catalysts. The goal of the study was to develop models that could be used to screen large virtual libraries of catalysts in order to suggest candidates for further synthesis and screening. The descriptors used to represent the catalysts did not require detailed information about the catalysts themselves; they could be calculated using only the topology of the ligands. Using an initial set of five descriptors, model accuracies of about 70% were observed from each learning algorithm. A larger descriptor set (with ten descriptors) allowed bag classifiers that were 80% accurate to be built. All models were carefully evaluated to detect overfitting (memorization of the training data) and one example of the effects of overfitting is provided. Because the descriptors used in this study can be calculated very rapidly and the models themselves are very efficient, these bag classifiers are well suited to screening very large virtual libraries.

KNN
支持向量机

Keywords: combinatorial chemistry, machine learning, catalysis

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Industrial and scientific interest have driven enormous amounts of experimental and theoretical research into polymer catalysis (Mülhaupt 2003, Maiti *et al* 2000, Rappé *et al* 2000, Gladysz 2000, Angermund *et al* 2000, Woo *et al* 1999). Despite all that has been learned, catalysis is still very much an unsolved problem: under most circumstances it is impossible to design effective new catalysts from first principles, either using a computer or on a piece of paper. As with the rest of chemistry, experimental search and refinement remains an integral part of catalyst discovery.

Though first-principles design of a new catalyst may not yet be feasible, we can borrow an approach from the field of drug discovery and use computational models, built from existing experimental data, in a decision-support role to accelerate the discovery process. In this mode of operation, a predictive model is used to screen huge collections of potential catalysts *in silico*. Machine-learning algorithms are ideally suited to develop the highly efficient models required for this virtual-library approach. Machine learning has been applied successfully to the prediction of a variety of material properties including superconductivity (Landrum and Genin 2002), ferromagnetism (Landrum and Genin 2003), structure prediction (Villars *et al* 2000) and heterogeneous catalysis (Ioffe 1988). There are also numerous examples of the application of learning methods to organometallic compounds; these include conformational analysis (Beyreuther *et al* 1996) and mining crystal-structure databases (Cundari *et al* 2000, 2002, Cundari and Russo 2001). The major stumbling block when applying machine-learning approaches is that they are hungry for data: a large collection of consistent experimental measurements is required to be able to develop a useful model. In recent years, the application of high-throughput (or combinatorial) methods to molecular catalysis (Murphy *et al* 2002, Wennemers 2001, Boussie *et al* 2003) has begun to produce data sets which are large enough to allow the application of machine-learning algorithms to the prediction of homogeneous catalyst properties.

0957-0233/05/010270+08$30.00 © 2005 IOP Publishing Ltd Printed in the UK

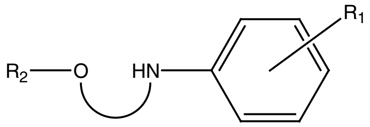


Figure 1. General form of the ligands used in this study.

For this work, we use a combinatorial catalysis data set published by a group from Symyx Technologies (Boussie *et al* 2003). The original paper contains results from both a primary and a secondary screen; here we use the secondary screening data. This data set consists of a set of 96 related ligands, the general form of which is sketched in figure 1.

These ligands were combined with $Hf(CH_2Ph)_4$ and an activator in solution to form catalysts that were then used to polymerize a mixture of ethylene and 1-octene. Four different values were recorded for each of the resulting polymers: yield, molecular weight (MW), polydispersity and percentage of 1-octene incorporated into the polymer. These data, consisting of almost 100 consistently collected activity values for homogeneous catalysts, are an ideal starting point for building predictive models.

In this contribution we discuss the descriptors used to characterize the ligands, provide a brief overview of the machine-learning algorithms applied and then present the results of building predictive models for the molecular weight of the polymer produced by each catalyst.

2. Methods

2.1. Descriptors

When using machine-learning methods, chemical substances are represented using a set of descriptors instead of (or in addition to) their chemical identity. Descriptors can be drawn from experimental properties or computed *ab initio*. They can be as complex as the HOMO–LUMO splitting of a molecule or as simple as its molecular weight. The most important constraint upon a set of descriptors used for learning is that, in order to build useful models, they must capture the relevant physics and chemistry of the problem. Thus, although machine-learning techniques themselves are problem independent, choosing the descriptors required to apply them to real problems requires significant domain knowledge.

In this study we require that the descriptors be inexpensive to compute. Although the data set used here is small enough to include time-consuming descriptors such as those from quantum-mechanical calculations, doing so would greatly limit the size of the virtual libraries that could be screened. Fortunately, the computational medicinal-chemistry community has developed a plethora of powerful and efficient descriptors to choose from. Because the only information we know for certain about the catalysts in this data set is the identity of the ligand (in most cases the active catalysts have not been identified), we limit the descriptor set to those that can be calculated based solely upon the identity of the ligands. We will, however, make the assumption that the ether oxygen and aromatic amine groups (see figure 1) are important in determining the properties of the catalysts; these atoms will be singled out in our descriptor set.

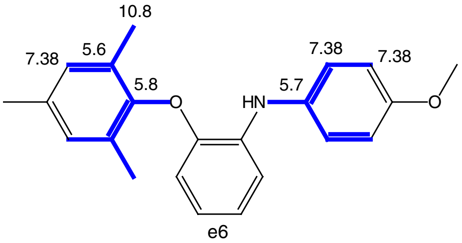


Figure 2. Contributions to the descriptors Surf3_N and Surf3_O for ligand e6.

Gasteiger charges, calculated by applying an iterative equalization scheme to orbital electronegativities, provide a very efficient source of approximate atomic charges in molecules (Gasteiger and Marsili 1980). Reasoning that the charges on the ligands' coordinating atoms will have a strong effect on catalyst properties, we include their Gasteiger charges as descriptors. Continuing this line of reasoning, the electrotopological state (EState) indices on the binding N and O, which can be viewed as approximating the electronegativity of the atoms in the molecule, were also used as descriptors (Kier and Hall 1999).

The molecular connectivity and shape descriptors developed by Hall and Kier (1991) provide powerful measures of molecular complexity and shape. After trying a variety of different Chi indices and Kappa values, Chi3N, Chi4N and Kappa3 were determined to be the most discriminating on this data set. A related descriptor that was found to be useful is BertzCT, an information-theoretic measure of molecular complexity (Bertz 1981).

Because the steric bulk of the ligands is known to play a role in the metal–ligand complexation efficiency in this system (Boussie *et al* 2003) and, in general, the degree of steric crowding around the metal atom can affect polymerization activity and polymer properties (Stehling *et al* 1994, Spaleck *et al* 1994), we would like to include some measure of sterics. In order to do so efficiently, we have developed a new class of descriptor: surface factors. The surface factor for a given atom is calculated by summing the contributions of neighbouring atoms that lie within a particular topological radius (number of bonds) of the molecule's approximate surface area (ASA). As a further refinement the bonds that lie on the path between the binding O and N are not considered—we assume that the steric bulk from these atoms will be 'behind' the catalytic metal centre and thus have a smaller impact on activity. Atomic contributions to the ASA are generated using the method developed by Labute (2000). The contributions to the surface factors of degree three for one of the ligands from our data set are illustrated in figure 2. These surface factors, calculated using topology and approximate contributions to the surface area of a ligand, are clearly too simple to capture the full complexity of the three-dimensional structure of the resulting catalysts. However, we will demonstrate that they contain enough information to allow useful predictive models to be built.

Our final descriptor set contains ten descriptors: Gast_O, Gast_N, EState_N, EState_O, Chi3N, Chi4N, Kappa3, BertzCT, Surf3_O and Surf3_N. These were calculated in a few seconds for the 96 ligands in the data set.

2.2. *Hierarchical clustering*

Hierarchical clustering refers to a class of algorithms that group data points into clusters based upon the distances between them in descriptor space. The procedure, in general, starts with a group of clusters made up of the individual data points. These clusters are recursively combined to form a tree (hierarchy) of clusters of increasing size. The tree is drawn with branch heights proportional to the distance between the joined clusters, see figure 6 for an example. The resulting cluster trees, which can reveal groupings and regularities in the data set, are often useful for analysing and understanding the data. A more detailed introduction to the theory and practice of hierarchical clustering can be found in Romesburg (1990) and Kumar (2000).

The descriptor values for the ligands were clustered using an Euclidean distance metric and Ward's minimum-variance method (Romesburg 1990). To help ensure that all descriptors were treated equally in the clustering, the data were standardized before being clustered: the data were shifted and scaled such that the average value and standard deviation of each descriptor were 0.0 and 1.0, respectively.

2.3. *Evaluating the quality of predictive models*

Any time one works with predictive (or statistical) models, it is important to be mindful of the famous quote from the statisticians George Box and Norman Draper: 'Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful' (Box and Draper 1987). In order to apply this wisdom, we must keep the intended purpose of our machine-learning models in mind and we need to be able to estimate how wrong they are. The models described in this paper are intended to be used for screening virtual libraries of potential catalysts. In order to be useful for this purpose, the models must be fast enough to allow us to quickly screen large numbers of catalysts and they must be accurate enough on *new compounds* (as opposed to compounds from the training set) to select interesting potential catalysts for the next round of synthesis. We would be building very different types of models if, for example, we were attempting to understand the mechanism of the polymerization reactions these compounds catalyse.

One of the largest risks encountered when building predictive models using small number of compounds is the loss of generalizability caused by overfitting, or memorization of the data (Hawkins 2004). This phenomenon causes the model to fit the training data highly accurately, but perform poorly on new data. For example, a tenth order polynomial will fit three points perfectly, but the resulting model is essentially useless for making new predictions. We will detect overfitting using two techniques: holdout data and shuffle tests.

The accuracy of a model on new data can be estimated by splitting the data set randomly into two pieces before model building. One piece, the *training set*, is used to build the model while the remainder, the *holdout set*, is used to test the performance of the model. By repeating the data split–model building–testing process multiple times and collecting statistics we can begin to get an approximate measure of the generalizability of the models being built. With the exception of the bag classifier, which has its own error metric, all modelling performance data presented here are averages of the performance from 50 separate random splits of the data into 80% training (77 catalysts) and 20% holdout (19 catalysts).

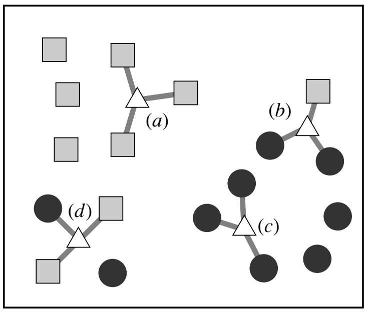


Figure 3. Illustration of the operation of a *k*-nearest neighbours model for $k = 3$. The classifications of the probe points (white triangles) are: (*a*) grey box, (*b*) black circle, (*c*) black circle, (*d*) grey box.

The risk of a model overfitting the data can also be assessed by randomly shuffling the activity values (leaving the overall distribution of activities unchanged) and rebuilding the model. The shuffling process removes all physical connection between the descriptors and activity, so the only possible way a shuffled model can perform well is by memorizing the data. If the shuffled models show performance comparable to that of the true models, overfitting is very likely a problem. Shuffle-test results reported here are an average of 50 different random shuffles of the data set.

2.4. *The k-nearest neighbours*

The *k*-nearest neighbours (KNN) is one of a family of 'neighbour-based' machine learning methods that examine a point's environment in descriptor space in order to classify it (Webb 2002, Mitchell 1997). In KNN, a point is classified by collecting votes from the *k* points in the training set that are closest to the probe point in descriptor space, figure 3.

Figure 3 also shows one of the potential pitfalls of using nearest-neighbour approaches: the classification of point (*d*) is very sensitive to its location in descriptor space. While point (*d*)'s current nearest neighbours are two grey boxes and one black circle, a small change in location alters the nearest neighbours to one grey box and two black circles, thereby changing the prediction.

2.5. *Decision trees*

Decision trees are a class of predictive models that classify data points by starting at the top of a tree—the root node—and moving down through the tree by asking a series of 'if-then-else' questions of the descriptor values at each branch point until a terminal (leaf) node is hit. Figure 8(*a*) shows an example. Due to their simplicity and clarity, decision trees lend themselves to interpretation. A good introduction to decision trees and the algorithms used to build them can be found in Mitchell (1997), while a recent survey of advanced techniques and the state of the art is found in Murthy (1998).

Decision trees are typically built using a *greedy* algorithm where the data are recursively divided into subsets based

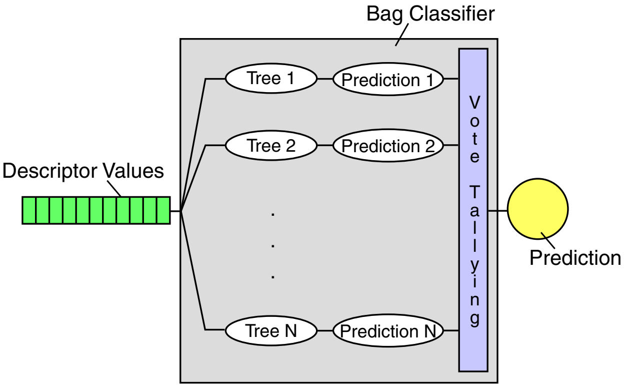


Figure 4. Illustration of a bag classifier using decision trees.

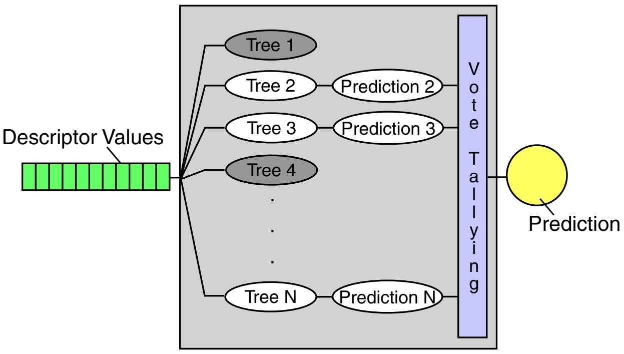


Figure 5. Illustration of the makeup of the classifier used in out-of-bag error estimation. Darkened trees, which saw the probe point in their training data, do not vote on the point's activity.

upon the descriptor which 'best classifies' it at each point. One standard approach, the ID3 algorithm, selects the best descriptor at each node using the concept of *information gain* introduced by Shannon as part of his development of information theory (Mitchell 1997, Shannon 1948a, 1948b). The decision trees presented here were generated using a modification of ID3 that allows real-valued descriptors to be used by selecting quantization bounds for each descriptor that maximize the information gain. This approach is similar to the methods described in Fayyad and Irani (1992, 1993).

2.6. Support vector machines

Support vector machines (SVMs) are a more sophisticated machine-learning technique that attempts to find an optimal hyperplane (linear separator) to classify the data. The data are often transformed to a higher dimension space (compared to the descriptor space) using kernel functions in order to deal with nonlinearity. The name of the method is derived from the 'support vectors'—points close to the decision boundary in feature space (Cristianini and Shaw-Taylor 2000, Burges 1998). Arising from the field of statistical learning theory, SVMs have rigorous theoretical underpinnings and, despite their great flexibility, are provably resistant to overfitting (Vapnik 2000, Cherkassky and Mulier 1998). The SVMs presented here were constructed using the LIBSVM software (Chang and Lin 2001) and were built using standard parameters.

2.7. Ensembles of learners: bag classifiers

There is an extensive body of literature documenting the power of using ensemble approaches—collections of individual models—for learning from real data (Webb 2002, Mitchell 1997, Dietterich 1997). Ensemble approaches have been demonstrated to be resistant to overfitting and able to handle complex problems. We have previously demonstrated the power of applying ensembles to the prediction of a variety of materials properties (Landrum and Genin 2002, Landrum *et al* 2003). In this work, we apply one of the earliest ensemble approaches: *bagging* (Breiman 1996a). A bag classifier consists of a collection of independent classifiers, here decision trees, combined into an ensemble, figure 4. Predictions are made for a point by allowing each decision tree to vote on the point's class; the class with the most votes 'wins' and is provided as the prediction.

The individual trees that make up the bag classifier are built using different random splits of the training data. Because each tree is trained on a different data set, they can be (and typically are) different from each other. Situations where this is not the case, where many of the individual trees in the bag are identical, typically indicate that the individual trees capture all the information present in the data. In these cases, the additional complexity of the bag classifier is often not required.

In addition to providing great flexibility and a resistance to overfitting, one significant advantage of bag classifiers (as well as other ensemble-based approaches to building models) is that their generalization ability can be tested using the training data—no holdout set is required. This approach, known as 'out-of-bag error estimation', has been demonstrated to give good estimates of the generalization error of a bag classifier (Breiman 1996b). Out-of-bag estimation takes advantage of the fact that not all data are used to build each tree in the bag: when screening a point in the training set, only trees that have not previously seen that point are allowed to vote on its activity, figure 5. In effect, this approach allows us to use the entire data set as both the training and holdout sets.

This ability to train on the entire data set and still obtain a reliable estimate of generalization ability is very useful when working with small data sets.

3. Results

The activity endpoint we focus on here is the molecular weight of the polymer produced by each catalyst. Because we are using classification (as opposed to regression) models, the measured MW values are grouped into two bins with a dividing line at 100k. This segmentation leaves the data set roughly balanced—53 (55%) ligands produce low MW polymer versus 43 yielding high MW polymer.

3.1. Descriptor set 1

Because overfitting is a significant concern when learning from only 96 points, our initial descriptor set is fairly small, containing only five descriptors: Gast_O, Gast_N, Chi3N, Surf3_O, Surf3_N.

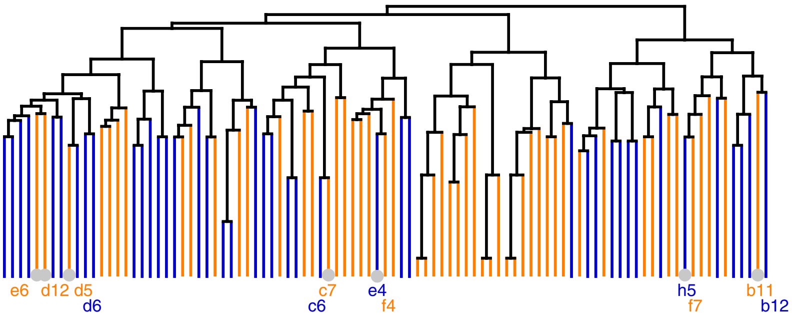


Figure 6. Hierarchical cluster tree for the homogeneous catalysis data set. Ligands (terminal points in the cluster tree) are shaded based upon the molecular weight of the polymer they produce: light for MW < 100 K, dark for MW ⩾ 100 K. Marked ligands are discussed in more detail in the text.

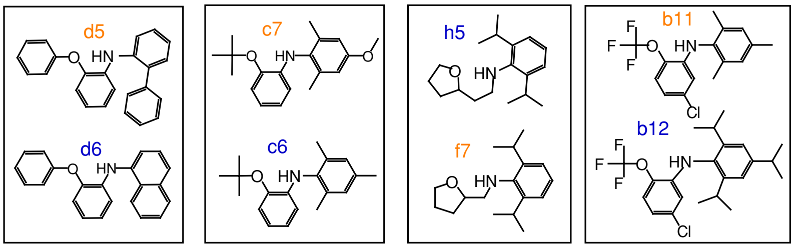


Figure 7. Examples of compounds with different activity values that appear together in the cluster tree shown in figure 6.

3.1.1. Clustering. The hierarchical cluster tree for our set of ligands is shown in figure 6, where ligand points have been coloured based upon the MW of the polymer they produce. It is immediately clear that points with similar activities tend to cluster together, indicating that our descriptor set is capturing much of the chemistry of the problem.

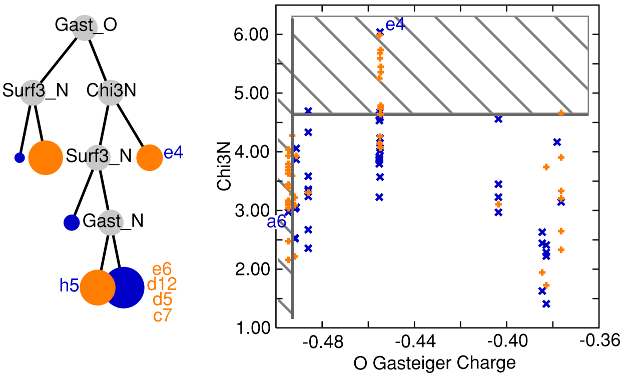
Reasoning that we can often learn more from the exceptions to a trend than from the trend itself, some of the exceptions to our 'like clusters with like' rule are shown in figure 7. The first thing we notice from these compounds is that each pair consists of ligands that are chemically quite similar to each other. For example, ligand c7 is derived from ligand c6 by replacing the *para*-methyl group with methoxy, a substitution that leads to a catalyst that produces polymers with much lower molecular weight (77k versus 210k). This suggests that descriptors encoding the electron-withdrawing/donating properties of ring substituents might be useful.

3.1.2. The k-nearest neighbours. In the hierarchical cluster tree of figure 6, ligands with similar activity were seen to cluster together. Because KNN is also based on distances in descriptor space, we expect reasonable performance from a KNN model. The results do not disappoint. The models built with $k = 3$ (this was determined to be the optimal value of k after trying a number of possibilities) are 79.2% (s = 3.4%) accurate across their training set and 68.1% ($s = 11.2$%) accurate on the holdout data. The sample standard deviation on the holdout set is large because the holdout set itself contains only 19 compounds; the s value observed roughly corresponds to a difference of only two misclassified compounds. The results show no significant bias—false negatives are about as likely as false positives. The reasonable holdout performance combined with the lack of bias, leads us to believe that overfitting is not a problem for this model. This conclusion is further reinforced by the shuffle test results: the shuffled KNN model is only 48.8% ($s = 12.1$%) accurate on its holdout set.

3.1.3. Decision trees. Seeking to improve upon the 68% holdout accuracy of the KNN models, we next built decision trees using descriptor set 1. The average accuracy of the decision trees on their training data was 83.1% (s = 2.5%). On the holdout data, this number decreased to 68.9% (s = 9.1%). Again, the relatively large sample standard deviations on the holdout data are to be expected because of the small size of the holdout data set. These trees are as accurate as the KNN models—classifying 13 of 19 compounds in the holdout set correctly on average, and unbiased—the six misclassifications are almost evenly split between high and low molecular weights. The shuffle test, which gives an average holdout accuracy of 48.0% ($s = 10.6$%) and shows a slight bias towards predicting low molecular weight, provides evidence that these results are not due to overfitting.

A representative decision tree is shown in figure 8(*a*). This tree, which is 74% accurate on the holdout set, correctly classifies 76 ligands overall. Decision trees operate by partitioning descriptor space into discrete regions and then assigning an activity value to each region. This is illustrated in the property plot in figure 8(*b*), which shows the segmentation of descriptor space by the first two levels of the decision tree.

Two broad regions of the Gast_O–Chi3N plane (crosshatched in figure 8(*b*)) are found to contain primarily ligands that produce low molecular weight polymers. Ligand a6, which appears to be misclassified in the property plot, is actually correctly classified by the next node of the tree (Surf3_N). Ligand e4, appearing at the top of the upper hatched region of the property plot, is misclassified by the tree.



(*a*) (*b*)

Figure 8. (*a*) A decision tree for predicting polymer molecular weight; terminal (leaf) nodes are scaled based upon the number of examples they classify. (*b*) the partitioning of descriptor space by the first two levels of the tree. Points and leaf nodes are coloured as in figure 3.

3.1.4. Support vector machines. Our next step upwards in model sophistication was to build support vector machines to generate models based on descriptor set 1. The SVMs had an average accuracy of 89.6% ($s = 5.4\%$) on their training set and 70.3% ($s = 9.8\%$) on the holdout data. Once again the error distribution showed no signs of bias, being approximately equally distributed between false negatives and false positives. We did not expect to see strong signs of overfitting in the SVMs, which are designed to be resistant to memorization. The shuffle test models' lack of accuracy, only 46.0% ($s = 10.9\%$) on their holdout data, is thus no surprise.

The SVMs have not, however, provided the increase in predictive accuracy we sought. The holdout accuracy of the SVMs is statistically indistinguishable from those of the KNN models and decision trees.

3.1.5. Bag classifiers. As a final test with descriptor set 1, the entire data set was used to build a bag classifier consisting of 50 decision trees. This process was repeated 50 times to collect statistics. The out-of-bag error estimate yields an average accuracy of 73.9% ($s = 2.5\%$). As with the other models, the results do not show significant bias. The out-of-bag accuracy of the classifiers built on shuffled data is 50.1% ($s = 5.55\%$), indicating that overfitting is unlikely to be a problem.

The performance of the bag classifiers is statistically indistinguishable from that of the support vector machines, but the bags do perform several per cent better than either the KNN models or the decision trees (the differences are significant at >99%). This difference suggests that bagging helps in improving the prediction accuracy. While 74% accuracy is good enough to enrich a virtual library, it is not good enough to leave us completely satisfied. Since we have not yet seen any signs of overfitting, there may be some room for improvement.

3.1.6. Error analysis. A statistical analysis of the models built using descriptor set 1 indicates that some ligands are consistently misclassified. A group of these ligands, each of which is missed by more than two thirds of the models, is highlighted in figure 6. Not surprisingly, these are the same compounds that were found to be very similar to ligands with differing activities. The descriptors being used do not characterize these compounds well enough to distinguish them from their neighbours.

Since essentially the same accuracy was obtained from models varying in complexity from KNN to bagged decision trees, we believe that it is unlikely that any learning algorithm will do much better using descriptor set 1. To obtain an increase in accuracy, the descriptor set must be expanded to deal with cases the current descriptor set is not capable of discriminating.

3.2. Descriptor set 2

Our second descriptor set is made up of the five descriptors from set 1 (Gast_O, Gast_N, Chi3N, Surf3_O, Surf3_N) along with five additional descriptors: Chi4N, Kappa3, EState_N, EState_O, BertzCT.

3.2.1. Decision trees. In order to illustrate one manifestation of overfitting in the results of a modelling experiment, decision trees were built using descriptor set 2. The average accuracy of the 50 trees across their training set was 99.9% ($s = 0.4\%$). This near-perfect accuracy fell to 63.3% ($s = 8.0\%$) on the holdout set. When compared to the decision trees built using descriptor set 1, performance on the training set went up more than 15% while holdout accuracy fell more than 5% (both differences are statistically significant at >99%). Adding descriptors *worsened* the generalizability of the model, a classic indicator of overfitting.

3.2.2. Bag classifiers. Because predictive models based on bagging tend to be more resistant to overfitting than the individual models of which they are composed, we next attempted to make use of the additional discriminating power of descriptor set 2 by building bag classifiers. Once again, bags containing 50 decision trees were built and tested. The average out-of-bag accuracy of the classifiers was 80.2% ($s = 2.5\%$), a substantial increase from the best accuracy using descriptor set 1 (73.9% with the bag classifier). The bag classifiers built on shuffled data had an out-of-bag accuracy of 62.9% ($s = 5.7\%$) with a substantial bias towards low MW predictions—more than 60% of the predictions were low MW. This bias, combined with the asymmetric distribution of the data (which is 55% low MW), explains part of the observed shuffle accuracy. However, it does appear that there is a risk of overfitting when using descriptor set 2, even using bagging.

Because the out-of-bag accuracy of the true models is far larger than that of the shuffled models, and because the true models do not demonstrate the bias exhibited by the overfit shuffled models, we believe that bagged models built from descriptor set 2 are reliable enough to make good predictions for virtual library screening. If higher accuracy were required, the current shuffle results indicate that *expanding* descriptor set 2 would not be the optimal way to achieve it; *replacing* some of the descriptors with better ones would be more likely to provide improvement without making the overfitting situation worse.

4. Conclusions

Using a set of very efficient descriptors to characterize the ligands, it was possible to build a variety of accurate models for the prediction of the molecular weight of polymer produced by a set of homogeneous catalysts. Despite the relatively small size of the data set, careful analysis and the use of shuffle tests and either holdout data or out-of-bag error estimates allowed us to determine that our predictions were not a result of overfitting the data. The descriptors were derived solely from the 2D structures of the ligands; no assumptions about the nature or structure of the catalysts themselves were required in order to develop these models.

Once they have been trained, all the models discussed here are extremely fast—the entire set of 96 catalysts can be classified in less than a second by any of the models. This very high throughput will ensure that the predictive models themselves do not become a bottleneck in virtual library screening.

To return to Box and Draper's quote: although our models are doomed to always be wrong, the high accuracy of the bag classifier, combined with the overall efficiency of the models and descriptors, leads us to conclude that they are indeed *useful*. We will be able to rapidly screen very large virtual libraries and suggest reasonable candidates for synthesis and screening.

Our future work in this area will focus on development of new descriptors to allow even more accurate models to be built. We will also develop models for the remaining activity endpoints. Preliminary results with descriptor set 2 and the bag classifier indicate that it is possible to build models for 1-octene incorporation that are more than 90% accurate.

Acknowledgments

This research was partially funded by an SBIR Phase I grant from the National Science Foundation: grant no DMI-Q20320242 'Library Design Algorithms for Active Learning in Drug Discovery'.

References

Angermund K, Fink G, Jensen V R and Kleinschmidt R 2000 Toward quantitatitive prediction of stereospecificity of metallocene-based catalysts for α-olefin polymerization *Chem. Rev.* **100** 1457–70

Bertz S H 1981 The first general index of molecular complexity *J. Am. Chem. Soc.* **103** 3599–601

Beyreuther S, Hunger J, Huttner G, Mann S and Zsolnai L 1996 Neural networks in conformational analysis *Chem. Ber.* **129** 745–57

Boussie T R *et al* 2003 A fully integrated high-throughput screening methodology for the discovery of new polyolefin catalysts: discovery of a new class of high temperature single-site group (IV) copolymerization catalysts *J. Am. Chem. Soc.* **125** 4306–17

Box G E P and Draper N R 1987 *Empirical Model-Building and Response Surfaces* (New York: Wiley-Interscience)

Breiman L 1996a Bagging predictors *Machine Learning* **24** 123–40

Breiman L 1996b *Out-of-bag estimation* ftp://ftp.stat.berkeley.edu/pub/users/breiman/OOBestimation.ps.Z

Burges C 1998 A tutorial on support vector machines for pattern recognition *Data Min. Knowl. Discovery* **2** 1–47

Chang C-C and Lin C-J 2001 *LIBSVM: A Library for Support Vector Machines* v2.5 http://www.csie.ntu.edu/∼cjlin/libsvm

Cherkassky V and Mulier F 1998 *Learning from Data: Concepts, Theory, and Methods* (New York: Wiley)

Cristianini N and Shaw-Taylor J 2000 *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods* (Cambridge: Cambridge University Press)

Cundari T R, Deng J, Pop H F and Sarbu C 2000 Structural analysis of transition metal beta-X substituent interactions. Toward the use of soft computing methods for catalyst modeling *J. Chem. Inf. Comput. Sci.* **40** 1052–61

Cundari T R and Russo M 2001 Database mining using soft computing techniques. An integrated neural network-fuzzy logic-genetic algorithm approach *J. Chem. Inf. Comput. Sci.* **41** 281–7

Cundari T R, Sarbu C and Pop H F 2002 Robust fuzzy principal component analysis (FPCA). A comparative study concerning interaction of carbon–hydrogen bonds with molybdenum–oxo bonds *J. Chem. Inf. Comput. Sci.* **42** 1363–9

Dietterich T G 1997 Machine learning research: four current directions *AI Mag.* **18** 97–136

Fayyad U M and Irani K B 1992 On the handling of continuous-valued attributes in decision tree generation *Mach. Learn.* **8** 87–102

Fayyad U M and Irani K B 1993 Multi-interval discretization of continuous-valued attributes for classification learning *13th Int. Joint Conf. on Artificial Intelligence*

Gasteiger J and Marsili M 1980 Iterative partial equalization of orbital electronegativity—a rapid access to atomic charges *Tetrahedron* **36** 3219–28

Gladysz J A 2000 Frontiers in metal-catalyzed polymerization: designer metallocenes, designs on new monomers, demystifying MAO, metathesis déshabillé *Chem. Rev.* **100** 1167–8

Hall L H and Kier L B 1991 The molecular connectivity chi indexes and kappa shape indexes in structure–property modeling *Reviews of Computational Chemistry* ed K B Lipkowitz and D B Boyd (New York: VCH Publishers)

Hawkins D M 2004 The problem of overfitting *J. Chem. Inf. Comput. Sci.* **44** 1–12

Ioffe I I 1988 *Application of Pattern Recognition to Catalytic Research* (New York: Wiley)

Kier L B and Hall L H 1999 *Molecular Structure Description: The Electrotopological State* (San Diego: Academic)

Kumar V 2000 *An Introduction to Cluster Analysis for Data Mining* http://www-users.cs.umn.edu/∼han/dmclass/cluster_survey_10_02_00.pdf

Labute P 2000 A widely applicable set of descriptors *J. Mol. Graph. Model.* **18** 464–77

Landrum G A and Genin H 2002 The rational discovery framework: a novel tool for computationally guided high-throughput discovery *Mat. Res. Soc. Symp. Proc.* **700** S7.6

Landrum G A and Genin H 2003 Application of machine-learning methods to solid state chemistry: ferromagnetism in transition metal alloys *J. Solid State Chem.* **176** 587–93

Landrum G A, Penzotti J E and Putta S 2003 Machine-learning models for high-throughput materials discovery *225th Am. Chem. Soc. Conf. (New Orleans, LA)*

Maiti A, Sierka M, Andezelm J, Golab J and Sauer J 2000 Combined quantum mechanics: interatomic potential function investigation of rac-meso configurational stability and rotational transition in zirconocene-based Ziegler–Natta catalysts *J. Phys. Chem.* A **104** 10932–8

Mitchell T 1997 *Machine Learning* (New York: McGraw-Hill)

Mülhaupt R 2003 Catalytic polymerization and post polymerization catalysis fifty years after the discovery of Ziegler's catalysts *Macromol. Chem. Phys.* **204** 289–327

Murphy V *et al* 2002 High-throughput approaches for the discovery and optimization of new olefin polymerization catalysts *Chem. Rec.* **2** 278–89

Murthy S K 1998 Automatic construction of decision trees from data: a multi-disciplinary survey *Data Min. Knowl. Discovery* **2** 345–89

Rappé A K, Skiff W M and Casewit C J 2000 Modeling metal-catalyzed olefin polymerization *Chem. Rev.* **100** 1435–56

Romesburg H C 1990 *Cluster Analysis for Researchers* (Malabar: Krieger)

Shannon C E 1948a A mathematical theory of communication *Bell. System Tech. J.* **27** 379–423

Shannon C E 1948b A mathematical theory of communication *Bell. System Tech. J.* **27** 623–56

Spaleck W, Küber F, Winter A, Rohrmann J, Bachmann B, Antberg M, Dolle V and Paulus E F 1994 The influence of aromatic substituents on the polymerization behavior of bridged zirconocene catalysts *Organometallics* **13** 954–63

Stehling U, Diebold J, Kirsten R, Röll W, Brintzinger H-H, Jüngling S, Mülhaupt R and Langhauser F 1994 *ansa*-zirconocene polymerization catalysts with annelated ring ligands—effects on catalytic activity and polymer chain length *Organometallics* **13** 964–70

Vapnik V N 2000 *The Nature of Statistical Learning Theory* (New York: Springer)

Villars P *et al* 2000 Interplay of large materials databases, semi-empirical methods, neuro-computing and first principle calculations for ternary compound former/nonformer prediction *Eng. Appl. Artif. Intell.* **13** 497–505

Webb A 2002 *Statistical Pattern Recognition* (Hoboken, NJ: Wiley)

Wennemers H 2001 Combinatorial chemistry: a tool for the discovery of new catalysts *Comb. Chem. High Throughput Screen* **4** 273–85

Woo T K, Margl P M, Deng L, Cavallo L and Ziegler T 1999 Towards more realistic computational modeling of homogeneous catalysis by density functional theory: combined QM/MM and *ab initio* molecular dynamics *Catalysis Today* **50** 479–500