# Virtual Screening Using Binary Kernel Discrimination: Effect of Noisy Training Data and the Optimization of Performance

Beining Chen,[†] Robert F. Harrison,[‡] Kitsuchart Pasupa,[‡] Peter Willett,*[,§] David J. Wilton,[§] and David J. Wood[†,§]

Departments of Chemistry, Automatic Control and Systems Engineering, and Information Studies, University of Sheffield, Sheffield S10 2TN, U.K.

Xiao Qing Lewell

GlaxoSmithKline Research and Development, Gunnels Wood Road, Stevenage SG1 2NY, U.K.

Received December 21, 2005

Binary kernel discrimination (BKD) uses a training set of compounds, for which structural and qualitative activity data are available, to produce a model that can then be applied to the structures of other compounds in order to predict their likely activity. Experiments with the *MDL Drug Data Report* database show that the optimal value of the smoothing parameter, and hence the predictive power of BKD, is crucially dependent on the number of false positives in the training set. It is also shown that the best results for BKD are achieved using one particular optimization method for the determination of the smoothing parameter that lies at the heart of the method and using the Jaccard/Tanimoto coefficient in the kernel function that is used to compute the similarity between a test set molecule and the members of the training set.

## INTRODUCTION

There is much interest in the use of machine learning methods for virtual screening in lead-discovery programs, using techniques such as substructural analysis (SSA)[1,2] and support vector machines (SVM).[3,4] In this study, we focus on a further machine-learning method known as binary kernel discrimination (hereafter BKD). BKD was first used in chemoinformatics by Harper and his colleagues[5,6] and is based on the calculation of similarities, via kernel functions, for chemical applications. The kernel function suggested by Harper[5] is

$$K_{\lambda}(i, j) = [\lambda^{n-d_{ij}}(1-\lambda)^{d_{ij}}]^{\beta/n} \qquad (1)$$

In (1), $\lambda$ is a smoothing parameter the value of which is to be determined, $n$ is the length of the binary fingerprints, $d_{ij}$ is the Hamming distance between the fingerprints for molecules $i$ and $j$, and $\beta$ ($\beta \leq n$) is a user-defined constant. Training set molecules are ranked using the scoring function

$$S_{\mathrm{BKD}}(j) = \frac{\sum_{i \in \mathrm{active}} K_{\lambda}(i, j)}{\sum_{i \in \mathrm{inactive}} K_{\lambda}(i, j)}, \qquad (2)$$

with the optimum value of $\lambda$ being found from analysis of the training set. The optimum is obtained by computing scores for each training set molecule using the other training set molecules for a number of different values of $\lambda$ in the range 0.50−0.99. For each value of $\lambda$ the sum of the ranks of the active molecules is computed. If this is plotted against $\lambda$ a clear minimum should be observed indicating the optimum $\lambda$, i.e., the value that minimizes the summed ranks of the actives in the training set. This optimum value is then used for scoring the molecules in the test set.

We have previously carried out several investigations of the use of BKD for virtual screening in both exact[7,8] and approximate[9,10] forms and have found that it provides an effective tool for ranking a database of previously untested molecules in order of decreasing probability of activity. Here, we discus the practical implementation of the method. First, we investigate the robustness of BKD in the face of noisy training data, a factor that is of great importance in operational contexts where there may be significant errors in the experimental data resulting from high-throughput screening (HTS) programs. We also compare the robustness of BKD with that of SSA. We then discuss two characteristics of BKD that might affect its virtual-screening performance: the procedure that is used to optimize the parameter $\lambda$ during the training phase of the method and the similarity coefficient that is used in the kernel function.

## EXPERIMENTAL DETAILS

All of the experiments reported in this paper used structural and biological activity data from the *MDL Drug Data* Report (MDDR) database,[11] specifically 102 535 structures and 11 activity classes that have been studied previously by Hert et al.[9] and by Bender et al.[12] These classes are summarized in Table 1. The molecules were represented by ECFP_4 fingerprints from the Scitegic Pipeline Pilot software system.[13] These encode circular 2D substructures of diameter

* Corresponding author phone: 0044-114-2222633; fax: 0044-114-2780300; e-mail: p.willett@sheffield.ac.uk.

[†] Department of Chemistry, University of Sheffield.

[‡] Department of Automatic Control and Systems Engineering, University of Sheffield.

[§] Department of Information Studies, University of Sheffield.

10.1021/ci0505426 CCC: $33.50
Published on Web 02/21/2006

**Table 1.** MDDR Activity Classes Used in the Study[a]

| activity class | active compounds | mean similarity |
|---|---|---|
| 5HT3 antagonists | 752 | 0.175 |
| 5HT1A agonists | 827 | 0.166 |
| 5HT reuptake inhibitors | 359 | 0.153 |
| D2 antagonists | 395 | 0.173 |
| renin inhibitors | 1130 | 0.337 |
| angiotensin II AT1 antagonists | 943 | 0.269 |
| thrombin inhibitors | 803 | 0.211 |
| substance P antagonists | 1246 | 0.179 |
| HIV protease inhibitors | 750 | 0.225 |
| cyclooxygenase inhibitors | 636 | 0.131 |
| protein kinase C inhibitors | 453 | 0.141 |

[a] "Mean similarity" denotes the mean pairwise similarity averaged across all of the activity class calculated using the Tanimoto coefficient and ECFP_4 fingerprints.

four bonds in which each atom in a molecule is represented by a string of extended connectivity values that are calculated using a modified Morgan algorithm. The Scitegic software represents a molecule by a list of integers, each describing a molecular feature and each in the range $-2^{31}$ to $2^{31}$; in our experiments, the integers describing a molecule were hashed to a fingerprint of length 1024 bits to give a fixed-format molecular representation.

In each of the experiments reported below, the test set was ranked using BKD, and a threshold applied to retrieve the molecules in the top 1% or the top 5% of the ranking. These molecules were then checked to determine the percentage of the active molecules that had been ranked above the threshold.

## EFFECT OF NOISY TRAINING DATA

HTS is an integral part of the lead generation stage of drug discovery but suffers from the low quality of much of the biological data that is created and the complexities incurred by the pooling of compounds.[14−17] It is hence most important that a virtual screening method is sufficiently robust that it can handle noisy data. Harper et al. found that BKD was more successful than neural network and similarity approaches when applied using moderately noisy input data (∼40% false positives, or FPs),[6] while Glick et al. have demonstrated that a Naïve Bayes Classifier (NBC) suffered only a minimal loss in accuracy when up to 80% of the 'hits' in a training set were actually FPs.[16] In this section, we evaluate the effectiveness of BKD when faced with noisy training data typical of that obtained from HTS. For comparison, we have carried out comparable experiments using substructural analysis (SSA).[1,2] Here, weights are assigned to each of the bits in a fingerprint, denoting that bit's probability of being set in active, as against inactive, molecules in the training set, and the score for a test set molecule is obtained by summing the weights for those bits that are set in that molecule's fingerprint. The experiments here used the R2 weight

$$\log\left(\frac{A_j/N_A}{I_j/N_I}\right)$$

where $A_j$ and $I_j$ are the numbers of active and inactive training set compounds with bit $j$ set, and where $N_A$ and $N_I$ are the total numbers of training set active and inactive compounds. This weight has been shown previously to perform well[2,18] and is closely related[10] to the NBC developed recently by Bender et al.[12] Experiments were carried out using both moderately noisy and extremely noisy input data, with the two groups of experiments using the 11 MDDR activity classes and ECFP_4 fingerprints. In all of the experiments reported in this section, BKD was run using a single, fixed value of 0.60 for the smoothing parameter, $\lambda$. Experiments were also carried out in which $\lambda$ was optimized for each training set; however, the results were on average no better than those obtained using this constant value and hence have not been included here.

**Table 2.** Comparison of Binary Kernel Discrimination (BKD) and Substructural Analysis (SSA) with Moderately Noisy Data[a]

| false positives (%) | 0% false negatives | | 10% false negatives | |
|---|---|---|---|---|
| | BKD | SSA-R2 | BKD | SSA-R2 |
| 0 | 82.3 | 59.6 | 75.2 | 58.6 |
| 20 | 72.1 | 58.8 | 63.2 | 58.1 |
| 33.3 | 63.0 | 57.3 | 54.0 | 56.3 |
| 50 | 48.3 | 54.2 | 38.4 | 52.4 |
| 60 | 37.9 | 50.1 | 26.5 | 46.7 |
| 66.7 | 28.0 | 45.5 | 18.6 | 39.2 |

[a] Mean percentage of actives retrieved in the top 1% of the ranking of the MDDR database when averaged over 11 activity classes and five training sets for each activity class.

Five initial, uncorrupted training sets were generated for each activity class, with each such training set containing 100 actives and 400 inactives. The actives were chosen from the MDDR so that none of them had a fingerprint-based Tanimoto similarity value greater than 0.55 with any of the other actives; the inactives were chosen at random from the MDDR. All the remaining, unselected compounds formed the test set. Each training set was corrupted by labeling randomly selected proportions of the inactive compounds as active (i.e., a false positive, FP) and, to a lesser extent, selected proportions of the active compounds as inactive (i.e., a false negative, FN). Combinations of {0|25|50|100|150|200} inactive compounds were randomly selected to be relabeled as actives and {0|10} actives were randomly selected to be relabeled as inactives: this resulted in corruption levels of 0, 20, 33, 50, 60, or 67% for the FPs and 0 or 10% for the FNs. For each combination of variables (activity class, training set, and level of corruption), five corrupted training sets were randomly generated with the results averaged over these five sets of runs.

Table 2 lists the mean percentage of the maximum possible number of actives retrieved in the top 1% of the ranked MDDR database, averaged over the 11 activity classes. With the clean training data SSA-R2 performed poorly when compared to BKD, retrieving less than three-quarters of the actives retrieved by BKD. However, SSA-R2 is much more tolerant to the presence of FPs in the training data, its performance relative to BKD increasing steadily as the degree of noise increases. Thus, when 100 inactive training compounds are relabeled as actives (50% FPs), SSA-R2 retrieved nearly all of the actives that were retrieved with the clean training data (54.2% compared to 59.6%), whereas BKD retrieved less than two-thirds of the active compounds retrieved with the clean training data (48.3% compared to 82.3%); the difference is still more marked when 200 inactive training compounds were relabeled as inactive (67% FPs).

**Table 3.** Comparison of Binary Kernel Discrimination (BKD) and Substructural Analysis (SSA) with Extremely Noisy Data[a]

| false positives (%) | BKD | SSA-R2 |
|---|---|---|
| 0 | 80.2 | 58.9 |
| 20 | 73.1 | 58.5 |
| 40 | 65.3 | 57.4 |
| 60 | 52.9 | 55.2 |
| 70 | 45.1 | 53.3 |
| 80 | 34.5 | 50.4 |
| 90 | 19.8 | 43.4 |
| 95 | 9.8 | 30.5 |

[a] Mean percentage of actives retrieved in the top 1% of the ranking of the MDDR database when averaged over 11 activity classes and five training sets for each activity class.

SSA-R2 is also more tolerant to the presence of FNs. When using 10% FNs with 0% FPs, the average percentage of actives retrieved by SSA-R2 is reduced from 59.6% to 58.6%, where as the average percentage of actives retrieved by BKD is reduced from 82.3% to 75.2%, a much greater reduction. At the most extreme level of corruption, 66.7% FPs and 10% FNs, SSA-R2 retrieves over twice as many actives as does BKD. Table 2 provides mean results averaged over the 11 activity classes, but entirely comparable conclusions can be drawn if the data for individual activity classes are considered. We hence conclude that BKD is much more effective than SSA-R2 when provided with clean training data but is much less effective when there is substantial noise in the training data.

Further experiments were carried out to test BKD and SSA-R2 with extremely noisy training data, in which up to 95% of the ‘actives’ were actually FPs. Here, five training sets of 100 active and 4000 inactive compounds were generated for each of the activity classes using the same method as previously, and then varying numbers of inactive compounds {0|11|25|43|67|100|150|233|400|900|1900} were randomly selected to be relabeled as actives to give results for 0, 10, 20, 30, 40, 50, 60, 70, 80, 90, and 95% FPs; no FNs were used. The results averaged over the 11 activity classes and five different training sets for each activity class are shown in Table 3. It will be seen that SSA-R2 is remarkably tolerant to noise in the training data. For example, when 90% of the actives in the training set are FPs, SSA-R2 still retrieved almost three-quarters of the actives it was able to retrieve with completely clean training data and retrieved over twice as many actives as did BKD.

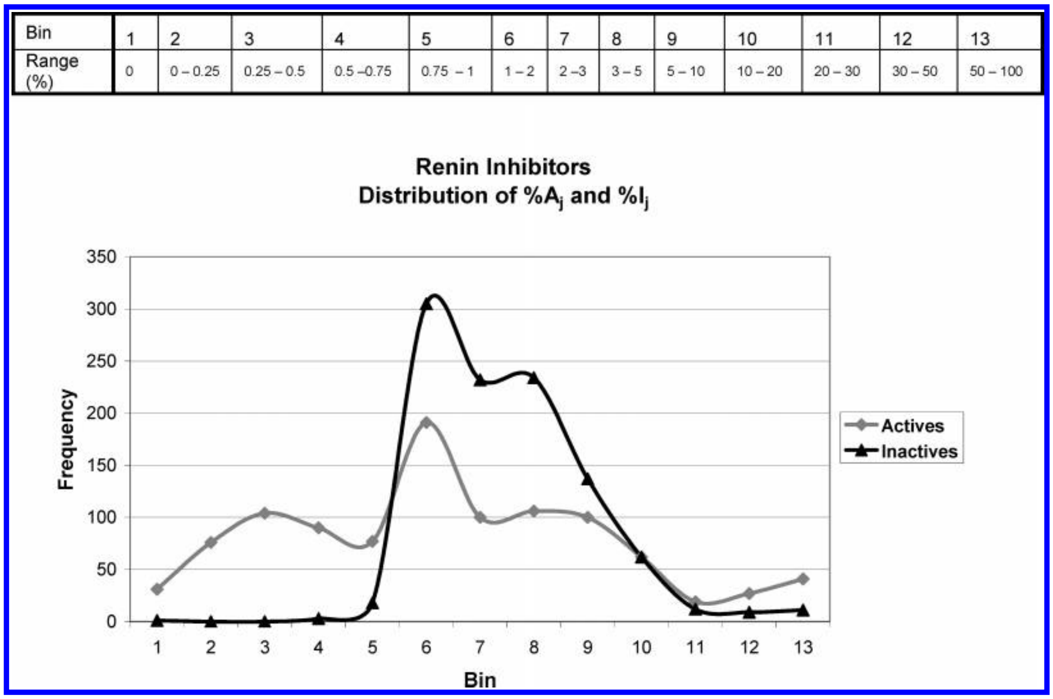
Given the marked differences apparent in Tables 2 and 3, we must ask why the two machine-learning methods differ so greatly in their ability to handle noisy data successfully. There appear to be two reasons for this behavior that, taken together, explain the limited robustness of BKD in our experiments.

The similar property principle states that similar molecules will have similar activities,[19,20] and we would hence expect that there would be at least some degree of structural commonality among the set of actives for a given activity class and also that the set of inactives for that class will be structurally disparate. If this is the case, then substructures (and hence bits in a fingerprint) will occur more randomly in a set of inactives than in a set of actives. This behavior is exemplified for the renin inhibitors in Figure 1.

For each of the 1024 bits in an ECFP_4 fingerprint, a note was made of the percentage of the active compounds, $\%A_j$, and the percentage of the inactive compounds, $\%I_j$, for which the $j$th bit was set; histograms were then generated to probe the differences in the two types of distribution over the 1024 bit positions in the fingerprint. This was achieved using the binning scheme shown in the upper portion of Figure 1 where it will be seen, for example, that Bin-10 denotes bits that were set in 10−20% of the actives (or 10−20% of the inactives). The resulting distributions for the molecules in the renin inhibitor activity class and for the corresponding inactives, i.e., the remainder of MDDR, are shown in the lower portion of Figure 1. If there are structural trends within the set of compounds we would expect a greater number of substructures to occur in a relatively high or a relatively low proportion of the compounds (the extreme ends of the histogram). If the set of compounds is structurally diverse we would expect more substructures to occur in a moderate proportion of the compounds as the set would have no tendency toward or against any particular substructure.

The distribution is more sharply peaked for the renin inactives, with a maximum in the sixth bin range (1−2%). This means that more than 300 of the 1024 bit-positions are set on 1−2% of the time for the renin inactives, as against less than 200 of the bit-positions in the actives, and indicates that the occurrence of the substructures is more evenly distributed in the inactive set. The distribution of the values of $\%A_j$ indicates that there are structural trends in the activity class and that while some fragments are over-represented in the actives (see bins 12 and 13) the trends often involve the absence of structural fragments, i.e., many of the bit positions in the actives are set less frequently than would be expected with a random distribution (see, e.g., the relative occupancies of bins 1−5). In a ‘clean’ training set, SSA identifies the fragments that are beneficial or detrimental to a molecule’s activity. If FPs are added to the training set, then the overall structural trends will be the same as the uncorrupted data set because the FPs are structurally diverse. Thus, the FPs introduced in our experiments will, in general, dilute the structural trends of the true positives (TPs), rather than changing the overall shape of the distributions. In practice, this means that the magnitudes of the various fragment weights are reduced as the training data becomes progressively corrupted but that the weights retain their sign. The reader should note that this explanation of the observed behavior assumes that there is no explicit structural relationship between the TPs and the FPs; there is no such relationship here since the FPs were generated at random, but this might not necessarily be the case in the laboratory.

Thus, SSA-R2’s robustness in the face of noisy data would appear to be due to the fact that it takes into account all of the molecules in the training set. This situation is rather different from BKD, where the scores are increasingly dominated by the largest kernel values as $\lambda$ is varied from 0.5 to 1.0, i.e., account is increasingly taken of just the most similar training set actives and inactives. Our experiments here have used a value of 0.60 for the smoothing parameter, and even this value is sufficiently large to ensure that the kernel values resulting from the closest single active or single inactive in the training set can dominate the summations that produce the final score, $S_{BKD}(j)$, for a test set compound. Compounds will be scored highly if they are a close neighbor of a single training set active and are relatively far away from any training set inactives, so that, in many cases, the



| Bin | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Range (%) | 0 | 0 – 0.25 | 0.25 – 0.5 | 0.5 –0.75 | 0.75 – 1 | 1 – 2 | 2 –3 | 3 – 5 | 5 – 10 | 10 – 20 | 20 – 30 | 30 – 50 | 50 – 100 |

**Figure 1.** Binning scheme for the values of %$A_j$ and %$I_j$ and resulting distributions for the frequencies with which bits are set, for the MDDR renin inhibitors and inactives.

**Table 4.** Kernel Function Values for the Five Closest Actives and the Five Closest Inactives in the Training Set for Two Test Set Molecules with $\lambda$ Set to 0.60

| rank position in test set | training set class | closest training set neighbors to the test set compound | | | | | sum of kernel functions | $S_{BKD}(j)$ |
|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | | |
| 1 | actives | 5.52E-01 | 1.29E-10 | 7.15E-11 | 5.86E-11 | 5.86E-11 | 5.52E-01 | 4.04E+07 |
| | inactives | 4.25E-10 | 1.92E-10 | 1.29E-10 | 1.29E-10 | 1.06E-10 | 1.37E-08 | |
| 1000 | actives | 4.21E-02 | 9.58E-07 | 1.09E-07 | 8.90E-08 | 7.30E-08 | 4.21E-02 | 5.39E+00 |
| | inactives | 7.09E-03 | 6.59E-04 | 4.12E-05 | 3.83E-06 | 2.11E-06 | 7.81E-03 | |

**Table 5.** Kernel Function Values for the Five Closest Actives and the Five Closest Inactives in the Training Set for Two Test Set Molecules with $\lambda$ Set to 0.52

| rank position in test set | training set class | closest training set neighbors to the test set compound | | | | | sum of kernel functions | $S_{BKD}(j)$ |
|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | | |
| 1 | actives | 8.55E-01 | 5.56E-01 | 4.95E-01 | 4.40E-01 | 2.75E-01 | 3.66E+01 | 1.78E-01 |
| | inactives | 2.75E-01 | 2.55E-01 | 2.35E-01 | 1.94E-01 | 1.94E-01 | 2.06E+02 | |
| 1000 | actives | 2.86E-01 | 2.75E-01 | 2.75E-01 | 2.01E-01 | 2.01E-01 | 2.55E+01 | 1.56E-01 |
| | inactives | 2.55E-01 | 2.55E-01 | 2.45E-01 | 2.45E-01 | 2.26E-01 | 1.63E+02 | |

ratio of the kernel scores obtained from the closest training set active and the closest training set inactive provides a reasonable approximation of the score, for the $j$th test set molecule.

Table 4 details the kernel contributions of the five closest training set actives and five closest training set inactives for the first-ranked and thousandth-ranked compounds, from a BKD run with a 5HT3 antagonist training set containing 80% FPs. The table also contains the sum of the kernel-functions (both active and inactive) and the overall BKD score for both of the molecules. It will be seen that the molecules are generally ranked because of their close proximity to a single active training set compound, which may well be an FP given the 80% corruption in this run.

The focus on a single nearest neighbor would suggest that we might obtain better results by setting a lower value for $\lambda$, so that proximity to a cluster of compounds could become more influential than close proximity to a single compound. This is illustrated in Table 5, which is of the same form as Table 4 but run with $\lambda$ set to 0.52. It will be seen that the sums of the kernel functions are much higher than the kernel contributions from any single training set compound, i.e., a much higher proportion of the training set is involved in the calculation of $S_{BKD}(j)$ for a test set compound. Tables 4 and 5 illustrate specific examples but the behavior is quite general. Thus, again using 5HT3/80% FPs data, define $R$ as the ratio of the kernel values for the first and second nearest neighbors and assume that a value in excess of 100 (i.e., 2 orders of magnitude) corresponds to the BKD score being dominated by the closeness to a single nearest neighbor. Then, $R$ for the actives was found to be in excess of 100 for 77.9% of the top 1% of the test set molecules when $\lambda$ was set at 0.60 but in excess of 100 for just 0.20% of these molecules when $\lambda$ was set to 0.52. The corresponding figures for the inactives are 9.10% and 0.00%.

The truly active compounds will share many structural features and so will occupy a relatively compact region of the descriptor space. Test set compounds that occur in this

**Table 6.** Effect of the Smoothing Parameter Value on BKD's Tolerance to Noise[a]

| false positives (%) | BKD-0.60 | BKD-0.53 | BKD-0.52 | SSA-R2 |
|---|---|---|---|---|
| 0 | 80.2 | 64.2 | 58.0 | 58.9 |
| 20 | 73.1 | 62.9 | 57.8 | 58.5 |
| 40 | 65.3 | 61.5 | 57.1 | 57.4 |
| 60 | 52.9 | 58.2 | 55.8 | 55.2 |
| 70 | 45.1 | 55.2 | 55.0 | 53.3 |
| 80 | 34.5 | 51.7 | 53.1 | 50.4 |
| 90 | 19.8 | 38.4 | 47.4 | 43.4 |
| 95 | 9.8 | 22.9 | 34.5 | 30.5 |

[a] Mean percentage of actives retrieved averaged over 11 activity classes and five training sets for each class.

active region of the descriptor space are scored highly owing to the proximity with the cluster of TP training set compounds. This effect is shown in Table 6, which compares the performance of BKD when used with values for $\lambda$ of 0.60, 0.53, and 0.52. It will be seen that lowering the value of the smoothing parameter improves BKD's tolerance to noisy training sets but at the expense of its performance when a clean training set is available: for example, Table 6 shows that with 60% FPs, SSA-R2 has overtaken BKD-0.60 but has itself been overtaken by BKD-0.52.

We hence conclude that the best results will be obtained with BKD if some information is available as to the level of noise in the training set. If the incidence of false positives is low, e.g., if all of the actives have been confirmed after retesting and/or after the determination of an IC50, then BKD will provide an effective virtual screening tool with the smoothing parameter, $\lambda$, set to about 0.60. Conversely, if only raw HTS data is available for training, then either a much lower value should be set for $\lambda$ or SSA-R2 should be used.

Very recently, a related study has been reported by Glick et al.[21] Building upon their previous work,[16] these authors studied the effect of added noise on the screening performance of an SVM, a recursive partitioning procedure, and an NBC that can be regarded as a form of SSA.[10] The study demonstrated that the SVM performed best with clean training sets but that the NBC was the most resistant of the methods to the effect of noise. It hence seems reasonable to conclude from the Glick et al. studies and from the work reported here that SSA is particularly appropriate when there is reason to believe that there is a substantial level of noise in the data available for training a machine-learning procedure.

## EFFECT OF THE OPTIMIZATION PROCEDURE AND THE SIMILARITY COEFFICIENT

There are several factors that might affect the virtual-screening performance of BKD. Two of these are the value of $\lambda$ that is used for the analysis of the test set and the coefficient that is used to measure the similarity between a pair of binary fingerprints in the kernel function. Here, we investigate the extent to which variations in these two factors could affect the performance of the method.

**Optimization of $\lambda$.** The parameter $\lambda$ lies at the heart of BKD, since it determines, effectively, the numbers of nearest neighbors that are used to predict the (in)activity of each test set molecule. Our studies thus far have optimized $\lambda$ using the approach suggested by Harper et al., which we refer to subsequently as the *sum of active rank positions* method and which was described in the Introduction to this paper. The approach is simple in concept, has low computational costs, and seems to work well; the experiments reported below were undertaken to see if it was possible to identify a superior approach.

The first modification was to use the *sum of the active rank positions squared*, the aim being to penalize more strongly parameter values that resulted in large numbers of poorly ranked active molecules. Next, we considered the *position of the median ranked active* (i.e., of the 50th highest active as there were 100 actives in the training sets used in these experiments) and the *position of the lowest ranked active* (i.e., of the 100th highest active). The former of these, based on the median, was not found to produce useful results as the position of the median active varied little with alterations in the value of the smoothing parameter; this method is hence not considered further. Finally, we considered two performance measures used previously by Ormerod et al. in a comparison of weighting schemes for substructural analysis.[2] In the ideal situation, all of the active compounds would appear before the inactive compounds; in practice, this is not achieved, and there will be some displacement of the active compounds. The *percentage misplaced score* for the ranked training set compounds is the percentage of inactive compounds that occur in the active section of the list, i.e., in the uppermost 50% of the ranking (and hence also the percentage of actives that occur in the lowermost 50% of the ranking). The *error score* adopts the same approach but additionally takes account of the positions of the misplaced compounds; specifically, each misplaced compound is scored by its distance from the perfect active/inactive threshold, and then the complete ranked list is scored by the sum of the distances of the misplaced compounds.[2]

It should be emphasized that all of these approaches are rather simple ways of choosing the best value for $\lambda$, and more sophisticated approaches have been described in the literature.[22,23] However, as Harper et al. note,[6] these are computationally more demanding and hence less easy to apply when large volumes of training data are available (as can often be the case in a corporate environment).

Training sets of 100 actives and 100 inactives were generated at random from each of the MDDR activity classes in turn. The value of $\lambda$ was varied from 0.50 to 1.0 in steps of 0.01, and the optimum value was selected using one of the scoring methods described above. This value of $\lambda$ was then used for the ranking of the test set, i.e., the remainder of the MDDR, and a note was made of the number of actives retrieved in the top 1% of the ranked database. Five different training sets were generated for each activity class, resulting in a total of 55 runs. The mean percentages of actives and mean $\lambda$ values (averaged over the five runs for each activity class and then over the 11 activity classes) obtained with each of the five scoring methods are listed in Table 7. Inspection of this table shows that the sum of the active rank positions and the sum of the active rank positions squared methods produce very similar percentages of actives (and optimum $\lambda$ values that were rarely further than 0.005 apart). The percentage misplaced also retrieved about the same number of actives, although the identities of these were frequently different from the two sum methods. The other two methods were noticeably inferior, with the error score

**Table 7.** Comparison of Procedures for Optimizing the Smoothing Parameter, $\lambda^a$

| activity class | sum of active rank positions | sum of active rank positions squared | lowest ranked active | error score | percentage misplaced |
|---|---|---|---|---|---|
| 5HT3 antagonists | 79.1 | 80.3 | 78.1 | 71.7 | 81.1 |
| 5HT1A agonists | 63.2 | 64.4 | 43.7 | 34.4 | 62.3 |
| 5HT reuptake inhibitors | 75.0 | 74.8 | 66.0 | 39.9 | 74.0 |
| D2 antagonists | 73.4 | 73.4 | 57.8 | 41.7 | 74.6 |
| renin inhibitors | 89.8 | 89.9 | 89.9 | 85.3 | 90.0 |
| angiotenisin II AT1 antagonists | 65.8 | 65.4 | 64.4 | 58.0 | 64.8 |
| thrombin inhibitors | 71.1 | 70.4 | 66.3 | 55.6 | 67.6 |
| substance P antagonists | 78.1 | 78.5 | 72.5 | 73.5 | 78.1 |
| HIV protease inhibitors | 80.4 | 80.6 | 80.0 | 72.0 | 81.9 |
| cyclo-oxygenase inhibitors | 64.7 | 63.3 | 54.1 | 31.6 | 65.3 |
| protein kinase C inhibitors | 83.7 | 83.8 | 83.6 | 72.5 | 83.5 |
| average mean percentage of actives | 74.9 | 75.0 | 68.8 | 57.8 | 74.8 |
| average $\lambda$ | 0.616 | 0.616 | 0.600 | 0.541 | 0.620 |

[a] The elements of the main body of the table contain the mean percentage of the actives in the top 1% of the ranked MDDR associated with that value, averaged over the five runs that were carried out in each case. The bottom row contains the mean value of $\lambda$ suggested as optimal by each method, averaged over the 11 MDDR activity classes and five runs for each activity class.

performing by far the worst, and consistently suggesting optimum values of $\lambda$ that were much less than for the other methods considered here (an average of 0.54 as against 0.60−0.62 for the other methods). There is hence no evidence here to suggest a change from the established sum of active positions method, a conclusion that is in agreement with recent work by Jorissen and Gilson on the parametrization of SVM for virtual screening.[3]

The significance of the results in Table 7 was tested using the Kendall Coefficient of Concordance ($W$), which is used to measure the degree of agreement between the rankings assigned to $N$ objects by each of $k$ judges.[24] Assume that $\bar{R}_i$ is the average of the ranks assigned to the $i$th object, then the test statistic, $W$, is given by

$$W = \frac{12\sum_{i=1}^{N} \bar{R}_i^{\,2} - 3N(N+1)^2}{N(N^2-1)}$$

(with a modified form of the equation being used when, as is often the case, two or more of the objects are assigned tied ranks). The significance of the computed value of $W$ can be obtained from standard tables for $N \leq 7$ or from the tables for the $\chi^2$ distribution with $N-1$ degrees of freedom for $N > 7$. In the context of Table 7, $k$ is 11 (the activity classes studied) and $N$ is 5 (the optimization procedures that are being compared); for the data in this table, $W$ is calculated to be 0.66, which is statistically significant ($p < 0.01$). Since a significant degree of correlation has been obtained, Siegel and Castellan suggest[24] that the best overall order of the $N$ objects is obtained by taking the mean ranks, $\bar{R}_i$, which suggests the following ordering:

sum of active rank positions squared >
sum of active rank positions = percentage misplaced >
lowest ranked active > error score

(although the results in Table 7 show that differences between the first three are very small indeed).

Experiments were carried out in which the test set was ranked using different values of $\lambda$ so as to identify the real optimal value, rather than the best value as suggested by one of the methods when applied to the training set. This optimal value varies between data sets, but in all cases it was noted that the average performance of BKD began to diminish rapidly when the value of the smoothing parameter was set to less than around 0.58, with the drop-off becoming increasingly pronounced once the value fell below about 0.56. It was noted that the sum of active rank positions method did, on some occasions, predict an optimum for $\lambda$ that was lower than this value, an observation suggesting a small modification in the overall optimization procedure. Specifically, if the sum of the active positions method suggests an optimum value for $\lambda$ that is lower than 0.60, then its optimum value is reset to this threshold value. The inclusion of this minimum threshold value for the smoothing parameter was found to yield an average improvement of about 1% in the percentage of actives retrieved using the sum of active positions method.

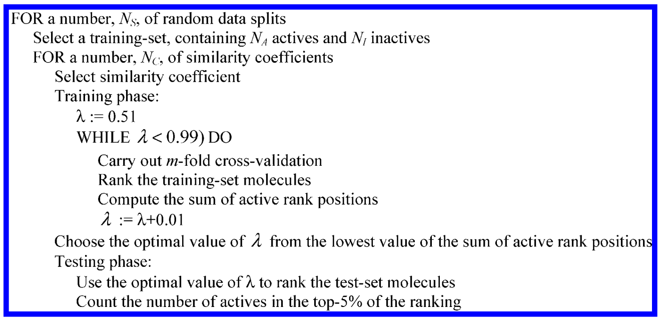
**Comparison of Similarity Coefficients.** The kernel function suggested by Aitchison and Aitken[23] in the original BKD paper is based on the number of bit positions at which two binary strings differ; this was adopted by Harper et al. in their application of BKD to virtual screening and has been used for all of the experiments reported thus far, both in this paper and in our previous work.[7−10] In principle, however, there is no reason one could not use an alternative measure of the distance (or similarity) between the fingerprints representing two molecules. Many different similarity coefficients are available for computing the similarities between pairs of binary vectors,[25] and comparative similarity-searching experiments with 2D fingerprints have suggested that the Tanimoto coefficient often performs best and that the Hamming distance is often markedly inferior. We have hence carried out a detailed series of experiments in which the Hamming Distance exponent, $d_{ij}$, in the kernel function, $K_\lambda(i,j)$, is replaced, as detailed in the Appendix, by one of the alternative similarity coefficients listed in Table 8.

It is worth noting that the score function, $S_{\mathrm{BKD}}(j)$, stands as surrogate for (and is directly proportional to) the estimated likelihood ratio (LR) for activity; the estimates being found via the Parzen Windows approach. It is convenient, though not essential, for such estimators that the kernel function is itself a density (probability mass) function. In particular, it

**Table 8.** Formulas for Similarity Coefficients for Binary (Dichotomous) Variables, Taken from the Review by Ellis et al.[26] [a]

| ID | common name | formula |
|---|---|---|
| A1 | Jaccard/Tanimoto | $S_{ij} = \frac{a}{a+b+c}$ |
| A2 | Dice | $S_{ij} = \frac{2a}{2a+b+c}$ |
| A3 | Russell/Rao | $S_{ij} = \frac{a}{n}$ |
| A4 | Simple Match | $S_{ij} = \frac{a+d}{n}$ |
| A5 | Hamman | $S_{ij} = \frac{a+d+b-c}{n}$ |
| A6 | Sokal/Sneath(1) | $S_{ij} = \frac{a}{a+2b+2c}$ |
| A7 | Sokal/Sneath(2) | $S_{ij} = \frac{2a+2d}{a+d+n}$ |
| A8 | Rogers/Tanimoto | $S_{ij} = \frac{a+d}{b+c+n}$ |
| A9 | Baroni-Urbani/Buser | $S_{ij} = \frac{\sqrt{ad}+a}{\sqrt{ad}+a+b+c}$ |
| A10 | Cosine/Ochiai | $S_{ij} = \sqrt{\frac{a+b}{a+c}}$ |
| A11 | Kulczyñski | $S_{ij} = \frac{\frac{a}{2}(2a+b+c)}{(a+b)(a+c)}$ |
| A12 | Simpson | $S_{ij} = \frac{a}{\min(a+b,a+c)}$ |
| C1 | Pearson | $S_{ij} = \frac{ad-bc}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$ |
| C2 | Yule | $S_{ij} = \frac{ad-bc}{ad+bc}$ |
| C3 | McConnaughey | $S_{ij} = \frac{a^2-bc}{(a+b)(a+c)}$ |
| D1 | Hamming distance | $D_{ij} = \frac{b+c}{n}$ |
| D2 | mean Euclidean | $D_{ij} = \frac{\sqrt{b+c}}{n}$ |
| D3 | Divergence | $D_{ij} = \frac{\sqrt{b+c}}{\sqrt{n}}$ |
| D4 | Bray/Curtis | $D_{ij} = \frac{b+c}{2a+b+c}$ |
| D5 | Soergel | $D_{ij} = \frac{b+c}{a+b+c}$ |

[a] In these formulas, it is assumed that two *n*-element binary vectors are being compared, one of which has *b* unique bits set, and the other of which has *c* unique bits set; *a* of these set bits are in common, and *d* bits are not set in either vector.



```
FOR a number, $N_S$, of random data splits
    Select a training-set, containing $N_A$ actives and $N_I$ inactives
    FOR a number, $N_C$, of similarity coefficients
        Select similarity coefficient
        Training phase:
            $\lambda := 0.51$
            WHILE $\lambda < 0.99$) DO
                Carry out $m$-fold cross-validation
                Rank the training-set molecules
                Compute the sum of active rank positions
                $\lambda := \lambda+0.01$
        Choose the optimal value of $\lambda$ from the lowest value of the sum of active rank positions
        Testing phase:
            Use the optimal value of $\lambda$ to rank the test-set molecules
            Count the number of actives in the top-5% of the ranking
```

**Figure 2.** Comparison of similarity coefficients. In these experiments: $N_S$ was set to 5, with each data split containing 20% of the total actives for the class; $N_A$ was chosen to be 10% of the total actives for the class, with $N_I$ having the same value; *m* was set to 5; and $N_C$ was 17, the number of distinct similarity coefficients tested.

should be non-negative and integrate (sum) to one over its domain. This latter condition may not be fulfilled for any given substitute for $d_{ij}$. However, for the purpose of ranking, all that is required is a score that is proportional to the estimated LR. It is straightforward to show that for any non-negative substitution for $d_{ij}$, the quantity computed via eq 2 remains directly proportional to the LR and is therefore a legitimate estimate.

The coefficients in Table 8 include association coefficients, $A_1$−$A_{12}$, correlation coefficients $C_1$−$C_3$, and distance coefficients, $D_1$−$D_5$. The definitions of these coefficients are taken from the review by Ellis et al.:[26] here, two *n*-element binary vectors are being compared, one of which has *b* unique bits set, and the other of which has *c* unique bits set; *a* of the set bits are in common, and *d* bits are not set in either vector. Of the 20 coefficients in Table 8, the Hamming distance, Bray/Curtis and Soergel have not been considered further as they are the complements of Simple Match, Dice, and Jaccard/Tanimoto, respectively, and thus give identical rankings; our experiments have therefore involved a comparison of the remaining 17 coefficients when applied to the 11 MDDR activity classes described previously, with the molecules again being characterized by ECFP_4 fingerprints.

The experimental design is detailed in Figure 2, and the results are in Table 9. Each element in this table contains the mean number of actives, when averaged over the five different data splits, retrieved in the top 5% of the ranked test set using the optimal value of $\lambda$ for a particular similarity coefficient. An inspection of this table shows that, while the mean values in the bottom row of the table cover only a limited range (83.0−88.2), there is a degree of variation in the performance of the various coefficients. Specifically, some of the coefficients—such as Jaccard/Tanimoto, Dice, and Sokal/Sneath(1)—provide a consistently high level of performance, while others—such as Russell/Rao, Simpson, Yule, and mean Euclidean—are consistently poor. The statistical significance of the level of agreement was tested using the Kendall W test (vide supra); the computed value of *W* was 0.45, which corresponds to a statistically significant $\chi^2$ value of 79.9 ($p \leq 0.01$ for 16 degrees of freedom). Given that a significant level of agreement between various rankings of the same set of objects has been established, the overall ranking of the 17 similarity coefficients is as follows:

$$\text{Jaccard/Tanimoto} > \text{Dice} > \text{Sokal/Sneath(1)} > \text{Pearson} > \text{Baroni-Urbani/Buser} > \text{Cosine/Ochiai} > \text{Kulczynski} \sim \text{McConnaughey} \sim \text{Divergence} > \text{Rogers/Tanimoto} > \text{Simple Match} \sim \text{Hamman} > \text{Sokal/Sneath(2)} > \text{Russell/Rao} > \text{mean Euclidean} > \text{Yule} > \text{Simpson}$$

The Hamming distance is the coefficient that is used in the basic kernel function, and it performed well in an earlier, but very limited, comparison of coefficients reported by Chen et al.[27] It gives exactly the same results as $A_4$, the Simple Match, and it will be realized that this coefficient is by no means the best of those studied here, coming in at the bottom half of the ranked list above. Instead, the best results are

**Table 9.** Comparison of Similarity Coefficients[a]

| | similarity coefficient | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| activity class | $A_1$ | $A_2$ | $A_3$ | $A_4$ | $A_5$ | $A_6$ | $A_7$ | $A_8$ | $A_9$ | $A_{10}$ | $A_{11}$ | $A_{12}$ | $C_1$ | $C_2$ | $C_3$ | $D_2$ | $D_3$ |
| 5HT3 antagonists | 93.9 | 93.8 | 91.8 | 90.9 | 90.9 | 94.1 | 90.8 | 90.6 | 92.3 | 93.6 | 93.2 | 89.2 | 92.8 | 86.6 | 93.2 | 89.1 | 91.7 |
| 5HT1A agonists | 87.9 | 87.6 | 85.3 | 86.7 | 86.7 | 87.2 | 86.5 | 86.3 | 87.2 | 86.8 | 86.0 | 83.8 | 87.9 | 78.8 | 86.0 | 83.8 | 86.8 |
| 5HT reuptake inhibitors | 73.6 | 72.5 | 70.8 | 70.8 | 70.8 | 74.1 | 70.2 | 71.5 | 73.1 | 72.6 | 72.7 | 69.0 | 73.1 | 64.7 | 72.7 | 71.3 | 72.0 |
| D2 antagonists | 77.9 | 78.3 | 77.7 | 76.5 | 76.5 | 78.2 | 76.5 | 76.6 | 77.9 | 78.3 | 78.4 | 75.3 | 78.5 | 69.1 | 78.4 | 72.2 | 75.9 |
| renin inhibitors | 98.8 | 99.1 | 98.9 | 99.1 | 99.1 | 98.9 | 99.1 | 99.2 | 98.9 | 99.0 | 98.8 | 96.5 | 99.0 | 98.5 | 98.8 | 98.1 | 99.4 |
| angiotensin II AT1 antagonists | 97.7 | 97.4 | 98.4 | 98.8 | 98.8 | 98.0 | 98.8 | 98.7 | 97.3 | 97.4 | 96.6 | 97.4 | 97.0 | 98.5 | 96.6 | 98.4 | 98.7 |
| thrombin inhibitor | 94.2 | 94.0 | 89.5 | 93.9 | 93.9 | 93.8 | 93.9 | 94.1 | 94.2 | 93.9 | 93.9 | 87.1 | 93.6 | 92.6 | 93.9 | 88.4 | 93.8 |
| substance P antagonists | 93.7 | 93.8 | 92.4 | 92.2 | 92.2 | 93.9 | 92.3 | 92.2 | 93.0 | 93.4 | 93.0 | 90.0 | 93.2 | 90.8 | 93.0 | 90.5 | 92.3 |
| HIV protease inhibitors | 94.9 | 94.3 | 91.2 | 94.8 | 94.8 | 94.3 | 94.7 | 94.8 | 94.8 | 94.4 | 94.5 | 91.5 | 94.8 | 94.0 | 94.5 | 87.0 | 94.7 |
| cyclooxygenase Inhibitors | 76.2 | 75.7 | 72.8 | 68.9 | 68.9 | 76.1 | 69.5 | 70.3 | 73.5 | 75.5 | 75.5 | 69.1 | 75.3 | 64.0 | 75.5 | 70.3 | 70.3 |
| protein kinase C inhibitors | 81.3 | 81.1 | 75.8 | 78.6 | 78.6 | 81.0 | 78.6 | 78.8 | 80.6 | 80.5 | 79.9 | 76.9 | 80.8 | 75.6 | 79.9 | 75.7 | 78.9 |
| mean percentage of actives | 88.2 | 88.0 | 85.9 | 86.5 | 86.5 | 88.2 | 86.4 | 86.6 | 87.5 | 87.8 | 87.5 | 84.1 | 87.8 | 83.0 | 87.5 | 84.1 | 86.8 |

[a] Each element of the table contains the mean percentage of actives (averaged over the five data sets for each activity class) retrieved in the top 5% of the ranking of the MDDR database.

obtained with the Jaccard/Tanimoto coefficient, which is, of course, very well established for conventional similarity searching applications.

Analogous results to those shown in Table 9 were obtained in experiments that used 10 further activity classes from the MDDR database, but that had been chosen to be as structurally diverse as possible (as discussed by Hert et al.[10]). These latter experiments support the conclusion that Hamming distance is not the best choice although there the Dice coefficient marginally outperformed the Jaccard/Tanimoto coefficient. While these two coefficients are monotonic, they give different similarity values and can thus result in slightly different rankings when the various active and inactive coefficient values are summed during the calculation of the $S_{\mathrm{BKD}}(j)$ scores. It is of interest to note that in these diverse classes, which present a more difficult challenge for a virtual-screening method, not only do Jaccard/Tanimoto, Dice, and Sokal/Sneath(1) perform, on average, up to 8% better than Hamming distance, they never perform worse: this is not always the case in the more homogeneous activity classes of Table 1. We hence conclude that the Jaccard/Tanimoto coefficient (or, equivalently, the Soergel distance) is the most appropriate for inclusion in the kernel function of a BKD procedure for virtual screening.

## CONCLUSIONS

Several recent studies have evaluated the use of binary kernel discrimination (BKD) for virtual screening of chemical databases. Experiments with 11 activity classes drawn from the *MDL Drug Data Report* database demonstrate that the optimal value of the smoothing parameter, and hence the predictive power of BKD, is crucially dependent on the number of false positives in the training set. We have also shown that the best results are achieved for BKD using the sum of active ranks squared method for optimizing the smoothing parameter $\lambda$ and using the Tanimoto coefficient in the kernel function that is used to compute the similarity between a test set molecule and the members of the training set.

## ACKNOWLEDGMENT

We thank the following: the Biotechnology and Biological Sciences Research Council and GlaxoSmithKline for funding David Wood; Gavin Harper for advice; the referees for comments on an earlier draft of this paper; MDL Information Systems Inc. for provision of the *MDL Drug Data Report* database; and the Royal Society, Scitegic Inc., Tripos Inc., and the Wolfson Foundation for hardware, laboratory, and software support.

## APPENDIX: USE OF SIMILARITY COEFFICIENTS IN KERNEL FUNCTION

An association coefficient measures the similarity between a pair of objects, while a distance coefficient measures the distance. It is common to convert a distance coefficient $d_{ij}$ via its complement to obtain the corresponding association coefficient $s_{ij}$. Let $s_{ij} = 1 - d_{ij}/n$ then, substituting into the kernel function A1:

$$K_\lambda(i,j) = (\lambda^{n-d_{ij}}(1-\lambda)^{d_{ij}})^{\beta/n} = (\lambda^{ns_{ij}}(1-\lambda)^{n-ns_{ij}})^{\beta/n} = (\lambda^{s_{ij}}(1-\lambda)^{1-s_{ij}})^{\beta} = (1-\lambda)^{\beta}\left(\frac{\lambda}{1-\lambda}\right)^{\beta s_{ij}}$$

Substitution into (A2) gives the scoring function

$$S_{\mathrm{BKD}}(j) = \frac{\sum_{i \in \text{active}} \left(\frac{\lambda}{1-\lambda}\right)^{\beta s_{ij}}}{\sum_{i \in \text{inactive}} \left(\frac{\lambda}{1-\lambda}\right)^{\beta s_{ij}}}$$

## REFERENCES AND NOTES

(1) Cramer, R. D.; Redl, G.; Berkoff, C. E. Substructural Analysis. A Novel Approach to the Problem of Drug Design. *J. Med. Chem*. **1974**, *17*, 533−535.

(2) Ormerod, A.; Willett, P.; Bawden, D. Comparison of Fragment Weighting Schemes for Substructural Analysis. *Quant. Struct.-Act. Relat.* **1989**, *8*, 115−129.

(3) Jorissen, R. N.; Gilson, M. K. Virtual Screening of Molecular Databases using a Support Vector Machine. *J. Chem. Inf. Model.* **2005**, *45*, 549−561.

(4) Saeh, J. C.; Lyne, P. D.; Takasaki, B. K., Cosgrove, D. A. Lead Hopping Using SVM and 3D Pharmacophore Fingerprints. *J. Chem. Inf. Model*. **2005**, *45*, 1122−1133.

(5) Harper, G. The Selection of Compounds for Screening in Pharmaceutical Research. Ph.D. Thesis, University of Oxford, 1999.

(6) Harper, G.; Bradshaw, J.; Gittins, J. C.; Green, D. V. S.; Leach, A. R. Prediction of Biological Activity for High-Throughput Screening Using Binary Kernel Discrimination. *J. Chem. Inf. Comput. Sci*. **2001**, *41*, 1295−1300.

(7) Wilton, D.; Willett, P.; Lawson, K.; Mullier, G. Comparison of Ranking Methods for Virtual Screening in Lead-Discovery Programs. *J. Chem. Inf. Comput. Sci*. **2003**, *43*, 469−474

(8) Wilton, D.; Willett, P.; Delaney, J.; Lawson, K.; Mullier, G. Virtual Screening Using Binary Kernel Discrimination: Analysis of Pesticide Data. *J. Chem. Inf. Model*. **2006**, *46*, 471−477.

(9) Hert, J.; Willett, P.; Wilton, D. J.; Acklin, P.; Azzaoui, K.; Jacoby, E.; Schuffenhauer, A. Comparison of Fingerprint-Based Methods for Virtual Screening Using Multiple Bioactive Reference Structures. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 1177−1185.

(10) Hert, J.; Willett, P.; Wilton, D. J.; Acklin, P.; Azzaoui, K.; Jacoby, E.; Schuffenhauer, A. New Methods for Ligand-Based Virtual Screening: Use of Data Fusion and Machine Learning to Enhance the Effectiveness of Similarity Searching. *J. Chem. Inf. Model.* **2006**, *46*, 462−470.

(11) The *MDL Drug Data Report* database is available from MDL Information Systems Inc. at http://www.mdli.com

(12) Bender, A.; Mussa, H. Y.; Glen, R. C.; Reiling, S. Similarity Searching of Chemical Databases Using Atom Environment Descriptors: Evaluation of Performance. *J. Chem. Inf. Comput. Sci*. **2004**, *44*, 1708−1718.

(13) The Pipeline Pilot software is available from Scitegic Inc. at http://www.scitegic.com

(14) Spencer, R. W. High-Throughput Screening of Historical Collections: Observations on File Size, Biological Targets, and Diversity. *Biotechnol. Bioeng.* **1998**, *61*, 61−67.

(15) Hann, M.; Hudson, B.; Lewell, X.; Lifely, R.; Miller, L.; Ramsden, N. Strategic Pooling of Compounds for High-Throughput Screening. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 897−902.

(16) Glick, M.; Klon, A. E.; Acklin, P.; Davies, J. W. Enrichment of Extremely Noisy High-Throughput Screening Data Using a Naive Bayes Classifier. *J. Biomol. Screening* **2004**, *9*, 32−36.

(17) Diller, D. J.; Hobbs, D. W. Deriving Knowledge through Data Mining High-Throughput Screening Data. *J. Med. Chem.* **2004**, *47*, 6373−6383.

(18) Cosgrove, D. A.; Willett, P. SLASH: a Program for Analysing the Functional Groups in Molecules. *J. Mol. Graphics Modell.* **1998**, *16*, 19−32.

(19) Brown, R. D.; Martin, Y. C. 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.

(20) Martin, Y. C.; Kofron, J. L.; Traphagen, L. M. Do Structurally Similar Molecules Have Similar Biological Activities? *J. Med. Chem.* **2002**, *45*, 4350−4358.

(21) Glick, M.; Jenkins, J. L.; Nettles, J. H.; Hitchings, H.; Davies, J. W. Enrichment of High-Throughput Screening Data with Increasing Levels of Noise Using Support Vector Machines, Recursive Partitioning, and Laplacian-Modified Naive Bayesian Classifiers. *J. Chem. Inf. Model*. **2006**, *46*, 193−200.

(22) Tutz, G. An Alternative Choice of Smoothing for Kernel Based Density Estimates in Discrete Discriminant Analysis. *Biometrika* **1986**, *73*, 405−411.

(23) Aitchison, J.; Aitken, C. G. G. Multivariate Binary Discrimination by the Kernel Method. *Biometrika* **1976**, *63*, 413−420.

(24) Siegel, S.; Castellian, N. J. *Nonparametric Statistics for the Behavioral Sciences*, 2nd ed.; McGraw-Hill: Singapore, 1988.

(25) Willett, P.; Barnard, J. M.; Downs, G. M. Chemical Similarity Searching. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 983−996.

(26) Ellis, D.; Furner-Hines, J.; Willett, P. Measuring the Degree of Similarity between Objects in Text Retrieval Systems. *Perspect. Inf. Manage*. **1994**, *3*, 128−149.

(27) Chen, B.; Harrison, R. F.; Hert, J.; Mpanhanga, C.; Willett, P.; Wilton, D. J. Ligand-Based Virtual Screening Using Binary Kernel Discrimination. *Mol. Simul.* **2005**, *31*, 597−604.

CI0505426