Molecular Transformations as a Way of Finding and Exploiting Consistent Local QSAR

Robert P. Sheridan,*,† Peter Hunt,‡ and J. Chris Culberson§

Molecular Systems Department, RY50S-100 Merck Research Laboratories, Rahway, New Jersey 07065, Neuroscience Research Centre, Terlings Park, Eastwick Road, Harlow, Essex CM20 2QR, U.K., and Molecular Systems Department, WP53F-301 Merck Research Laboratories, West Point, Pennsylvania 19486

Received August 11, 2005

The idea of a "transformation", making a small change to a chemical structure, for instance removing or replacing a substituent, is familiar to chemists. We suggest two ways of representing a transformation in silico, as a substructure descriptor difference vector, and as the set of atoms remaining once a maximum common substructure is eliminated. Such transformations can be filtered sensibly, and it is easy to compare one transformation to another. These representations have two applications. First, we can use these methods to automatically organize and display sets of closely related compounds such that any consistent local QSAR in a data set can be easily seen, the T-ANALYZE application. Second, we can suggest to a chemist how to change a molecule "on hand" to a more active one based on local QSAR for that activity, the T-MORPH application.

INTRODUCTION

In medicinal chemistry programs investigators accumulate biological activities for a set of compounds, which may include many close analogues. There are various approaches to understanding the quantitative structure−activity relationships (QSAR) contained in the data, each with its own advantages and pitfalls. Consider the set of simplified compounds in Figure 1. Familiar statistical QSAR methods (PLS, SVM, random_forest, etc.) produce what can be called "global QSAR" in the sense that they summarize structure−activity over the whole set and are heavily influenced by the most active and the most inactive compounds. In our example, most QSAR methods will note that having an imidazole at a certain position (molecules **1** and **2**) is good for activity and having a carboxylate at that position (molecules **10** to **12**) is bad. On the other hand, because both Cl and Br are distributed among compounds spanning the entire activity range, most QSAR methods (especially the ones that assume linearity) will not notice that changing Br to Cl in otherwise identical compounds (**2** vs **1**, **6** vs **3**, etc.) always results in an increase in activity. The common approach of finding the smallest changes that give rise to the largest changes in activity also has its limits. For example one would note here that changing dimethylamino to methylamino (compounds **3** and **7**) causes a large loss in activity. The fact that the same change in compounds **8** and **9** makes little difference in activity would not be noticed because that method is not looking for small activity differences.

Comparison of related compounds, pairwise or in small numbers, is a technique regularly applied by most medicinal chemists, who try to define "local QSAR" trends with every new result they obtain. This sort of analysis would certainly identify the above Br to Cl change as being beneficial. Furthermore should a chemist's latest compound fall below expectations, he or she would seek out what change has caused the loss or what changes have been seen previously to correct such a deficiency. However, when the data set is large (say more than a few tens of compounds), identifying related compounds "by hand" and keeping track of what trends are consistent becomes an overwhelming and error-prone process, and the chemist can easily draw incorrect conclusions by looking at too small a sample of the data.

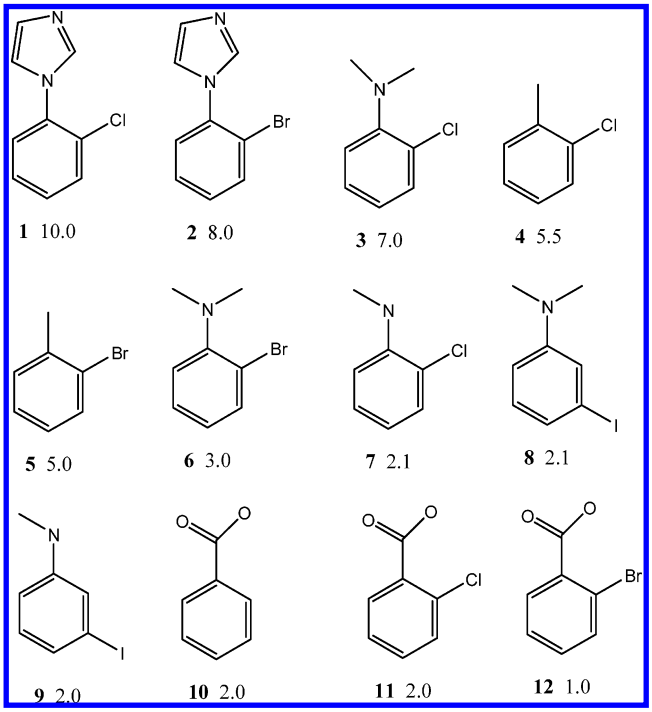


Figure 1. A small set of compounds to illustrate points about local- vs global-QSAR. The molecule name is in bold, followed by the activity.

* Corresponding author e-mail: sheridan@merck.com.
† RY50S-100 Merck Research Laboratories.
‡ Neuroscience Research Centre.
§ WP53F-301 Merck Research Laboratories.

10.1021/ci0503208 CCC: $33.50
Published on Web 11/03/2005

In this paper we demonstrate two methods which mimic the chemist's local QSAR approach, as opposed to the more common global QSAR approach of PLS etc. These methods automate the process of finding small changes in structure ("transformations") and how they relate to activity. Our implementation of the local QSAR methodology has given rise to two applications. The first is to take a large data set and organize it so that similar transformations are grouped together and the trends in activities can be easily perceived ("transformations analyze" or T-ANALYZE). The second application ("transformation morphing" or T-MORPH) is to address the question: "Given a compound on hand, what changes can I make to it to improve the activity based on the transformations within the data set?". We describe the application of these methods to three IC_{50}/Ki data sets: D2 agonists, dihydrofolate reductase inhibitors, and ACE inhibitors.

METHODS

There are two major applications: T-ANALYZE for organizing a data set and displaying the local QSAR within it and T-MORPH for improving the activity for a compound "on hand". Although they incorporate similar concepts, they have very different uses and therefore have different methodology in detail.

T-ANALYZE. An overview of the methodology for T-ANALYZE is as follows:

1. Start with a set of connection tables and corresponding activities for a data set.
2. Generate topological descriptors for the molecules.
3. Find pairs of molecules that are more similar than a user-defined cutoff.
4. For each pair above, define descriptor difference vectors A→B and B→A. These are "descriptor transformations". Find the maximum common substructure (MCS) of A and B and find the atoms that remain when the MCS is deleted. This is a "MCS transformation".
5. Compare pairs of transformations A→B and C→D to find those pairs that are "congruent".
6. Cluster the transformations based on their congruency in step 5. Eliminate redundant clusters.
7. Add activity data to the clusters. Sort the clusters and display to the user.

Note that the effort for the entire process can vary as steeply as N^4, where N is the number of molecules, because in step 5 we are examining "pairs of pairs". Therefore it behooves us to severely prune the A→B pairs we will consider before step 5 and also to prune the pairs of transformations we examine in step 5.

Details follow:

Step 2. One can think of each molecule as a vector of substructure descriptors and their frequencies. We use variants of the atom pair (AP)[1] and topological torsion (TT)[2] descriptors. AP's are of the form $Type_i$-$Type_j$-distance, where $Type_i$ and $Type_j$ are atom types and "distance" is the shortest through-bond distance between atoms *i* and *j*. TT's are of the form $Type_i$-$Type_j$-$Type_k$-$Type_m$ where atoms *i* through *m* are contiguously bonded atoms. Type includes element, number of non-hydrogen neighbors, and number of pi electrons. For instance type "C30" is a carbon with three non-hydrogen neighbors and zero pi electrons, e.g. −CH<. In some cases we want to take stereochemistry into account. In APC and TTC descriptors the atom types of chiral atoms are extended to include "R" or "S", e.g. C30R.

Step 3. The assumption made here is that to have meaningful comparisons, the transformation must involve changes that are small compared to the size of the molecule. One way to ensure this is to be sure that molecules being compared, A and B, are similar. Calculating the similarity based on substructure descriptors is much faster than calculating MCSs (in the next step), so this step is a useful screening function. We calculate similarity using the Dice index on substructure descriptors. The user sets the threshold above which the molecules are said to be similar, by default 0.7 for AP or 0.6 for TT. At these cutoffs molecules A and B will appear to be obvious analogues to most chemists. For the examples here we will use AP. Pairs of molecules that have identical descriptors, i.e., a similarity of 1.0, are ignored. In principle one can also select pairs of molecules based on whether their activities are "sufficiently different". For the T-ANALYZE application, we want to include transformations that involve no significant change in activity, so we generally ignore the activities at this stage.

Step 4. In this step, the descriptor-based transformation A→B will be the vector difference A minus B. For instance, Figure 2 shows such a vector difference for the (nonchiral) AP descriptors. In this application we store transformations for both directions A→B and B→A. We use the method described in Sheridan and Miller [3] to find the highest-scoring common substructure (HSCS) between the molecules A and B. This is a clique-based method that finds a set of corresponding atoms (the common substructure) from A and B such that the corresponding atoms have the same atom type and the same through-bond distances between them. The HSCS can consist of more than one fragment, but we typically use a "discontinuity penalty" of 1.0 to encourage the selection of HSCSs with the minimum number of fragments. Atom types for the purposes of finding the HSCS include element type, hybridization, and perhaps "R" or "S" depending on whether stereochemistry is to be taken into account.

The union of atoms from molecules A and B after the HSCS is removed is called the "remainder after elimination of common substructure" or RECS, which is our definition of the MCS-based transformation. The RECS is processed in two ways. First, we count the number of separate fragments in the MCS and the RECS. Only certain combinations are allowed, as shown in Figure 3. Disallowed combinations are usually those where A and B differ in more than one place. By default, we eliminate those A→B transformations because they are harder to interpret. Second, we calculate a hash string[4] summarizing the RECS. The hash depends only on the atoms in the RECS and their bonds; the order of the atoms is irrelevant, so the RECS hashes for A→B and B→A are identical. The hash string is important for the next step. Note that the context of the atoms in the RECS is lost in the MCS-transformation, but most of the context information is retained in the descriptor-based transformation.

Step 5. At this step we compare pairs of transformations A→B and C→D to see if they are "congruent". Congruent pairs are those that pass all of the following criteria:

a. A, B, C, and D are distinct molecules.

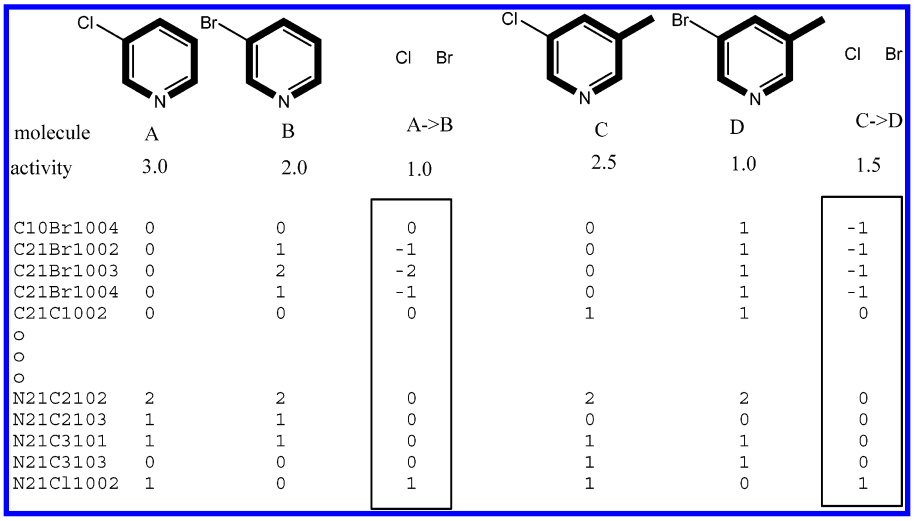


Figure 2. An illustration of two transformations A→B and C→D. The highest-scoring common substructure is in bold. Two ways of representing the transformations are shown: a descriptor-based transformation (calculated by subtracting AP descriptor frequencies) and a maximum common substructure-based transformation (by removing the common substructure).

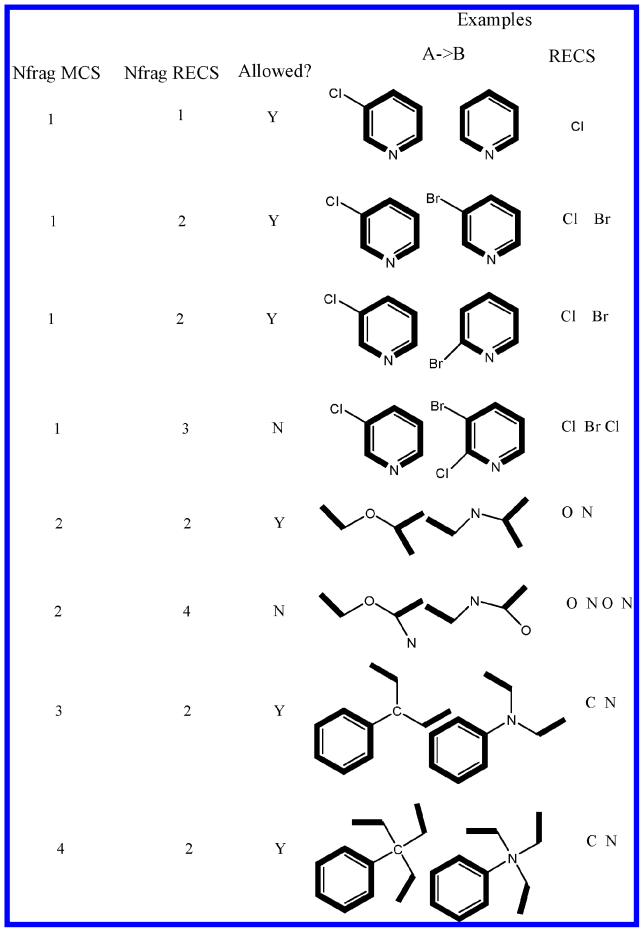


Figure 3. Acceptable transformations for the T-ANALYZE application based on the number of fragments in the maximum common substructure (MCS) and the remainder after eliminating the common substructure (RECS). Transformations where the molecules change in more than one place are considered uninteresting and are eliminated.

b. A, B, C, and D are all similar to each other, given a slightly loosened similarity criteria. We already know that A is similar to B and C is similar to D from step 3, say at the 0.7 cutoff for AP descriptors. But now we check that A is similar to C and D at 0.6. The same applies to B against C and D.

c. The descriptor-based vector A→B should have a normalized dot product to the C→D vector above a user-defined threshold, by default 0.7 using the AP descriptor, i.e., the descriptor-based changes are going roughly in the same direction.

d. The RECS hash for A→B exactly matches the hash for C→D.

In this implementation, the descriptor-based vector comparison in 5c provides the molecular context for the RECS in 5d.

Step 6. Now that we have decided which pairs of transformations are similar, we use the algorithm of Butina[5] to cluster them. This results in a set of non-overlapping clusters of various sizes. Note that since each vector-based transformation was generated in both directions, some clusters are nearly redundant with others. That is, a cluster can contain A→B and C→D and another cluster will contain B→A and D→C, which is exactly equivalent. Redundant clusters are eliminated.

Note at this point all our clusters have at least two transformations because of the congruence requirement. We now add "singleton clusters", A→B transformations for which there are no congruent transformations, i.e., the transformations that occur only once in the data set.

Step 7. At this point we can add activity information to each cluster, specifically the difference in activity associated with each transformation. For instance if we have a cluster consisting of the transformations A→B and C→D in Figure 2, the activity differences (Diff) for the individual transformations are 1.0 and 1.5, respectively. The mean difference ⟨Diff⟩ for this two-transformation cluster is the mean of 1.0 and 1.5 or 1.25. One can also calculate the "Agreement" of the cluster, i.e., what fraction of the time does the structural change move the activity in the same direction. This is calculated analogously with ⟨Diff⟩, except that one averages over *sign*(Diff): *sign*(Diff) = 1 if Diff >0, −1 if Diff < 0, 0 if Diff = 0. In this case the Agreement is 1.0; the two transformations both have a positive Diff.

The significance of ⟨Diff⟩ is not clear until we compare it with other clusters of the same size. We can randomly select two transformations that survived step 4 and calculate their ⟨Diff⟩. Repeat this 1000 times, and we will get a distribution

of mean differences that centers around zero and has some standard deviation. We can then calculate a "Z-score" for a two-transformation cluster relative to "random" two-transformation clusters as the ⟨Diff⟩ for the cluster on hand divided by the standard deviation. We can do this for clusters of any size (including the singletons). Z-scores can be positive or negative, depending on the arbitrary order of the molecules, i.e., A→B vs B→A. As the number of transformations per cluster grows, the standard deviation gets smaller exponentially, so for a given ⟨Diff⟩, the absolute value of the Z-score will be larger for larger clusters. Currently, we do not use the Z-score to determine whether a cluster is "statistically significant" in an absolute sense, only to sort the clusters (see below).

Clusters can be sorted in a number of ways depending on what aspects the user finds most interesting:

a. The absolute value of ⟨Diff⟩ emphasizes the changes in structure that result in the largest changes in activity.

b. The number of transformations in the cluster emphasizes the structural changes that are the most common in a data set.

c. The absolute value of the Z-score emphasizes the transformations that make a large difference in activity and simultaneously are more consistent among more examples. This is the default.

Clusters are displayed as sets of transformations, each transformation formed by a pair of molecules. For each molecule the structure and the name of the molecule are shown, with their individual activities. For every transformation, the difference in activity is displayed. For every cluster, we display the ⟨Diff⟩ and the Z-score. Within a cluster, it is arbitrary which way we display the pairs, i.e., the cluster A→B, C→D, E→F with $\langle Diff \rangle = 1.5$, $Z = 3.0$ could also be written as B→A, D→C, F→E, $\langle Diff \rangle = -1.5$, $Z = -3.0$. Our usual practice is to display each cluster with a positive ⟨Diff⟩, i.e., on average the first molecule in the transformation has a higher activity than the second.

T-MORPH. This is the overview for the T-MORPH application:

1. Start with a set of connection tables and corresponding activities for a data set.
2. Generate topological descriptors for the molecules.
3. Find pairs of molecules that are more similar than a user-defined cutoff.
4. For each pair above, define descriptor difference vectors A→B. Store the descriptor transformations in a database.
5. Calculate the descriptors for a probe molecule
6. Identify all transformations in the database that could apply to the probe molecule.
7. Cluster the applicable transformations.
8. Add the activity data, sort the transformations within each cluster, and present to the user.

Here the comparisons are linear with the number of transformations in the data set (e.g. N^2), there are many fewer comparisons which the user would have to inspect, so we can be more liberal in the types of transformations allowed. T-MORPH is fast enough to be used as an interactive Web-based application.

Steps 1−3 are the same as in the previous application.

Step 4. In this step, the descriptor-based transformation A→B will be the vector difference A minus B. For this application we store only one direction (e.g. A→B, but not B→A). This saves us space and reduces search times (important in a Web-based application). The reverse direction can be searched by negating the sign. We do not use the MCS transformation in this application.

Step 5. The user provides a probe molecule, for which AP and TT descriptors are calculated, a desired direction of change for the activity (i.e. increase or decrease) and a criterion for "equivalence" (i.e. a value for the difference in activity below which the two activities are deemed to be "the same").

Step 6. Here we check whether each transformation is compatible with the probe. The first test is to compare the frequency of all the *negative* TT's in the stored transformation to the frequency of occurrence of that descriptor in the probe molecule. If the probe contains more of these descriptors than is required to be removed by the transformation, then the transformation is deemed applicable to the probe. For example if the transformation is a pyridyl-Cl→pyridyl-Br change as shown with molecules A and B in Figure 4, then transformation A→B is encoded in the changes in the TT descriptors as summarized in the column within the blue box. If one examines the two probes E and F, then one can see that only probe E has both the TT descriptors (Cl10C31C21N21 and Cl10C31C21C21) which are required to be removed by the transformation A→B. Probe F does not contain the ClC31C21C21 descriptor (shown by the red 0 in the final column), and so the transformation is not applicable to this molecule. If the original transformation A→B is found to be not applicable, we negate it and try the B→A transformation. This corresponds to the descriptors enclosed in the dashed box in Figure 4; however, in this case both probes fail as they do not contain Br. TT's are used because they describe the 'local' effects and environment of a change. One consequence of this is that, unlike in T-ANALYZE, similar transformations will be clustered together regardless of whether the long-range environment of the transformations are similar, i.e., the transformations can apply to different chemical classes.

Step 7. Having retrieved all the transformations which could be applied to the probe, one then takes the complete transformation vector (AP's and TT's) and creates a self-similarity matrix, computed from the normalized dot product of each vector to every other vector. This matrix is then clustered using a hierarchical (average linkage) clustering method, and those clusters existing at a similarity height of 0.4 are reported to the user. The inclusion of AP's in this similarity calculation does help to bring some similarity to the molecules within a cluster but is not as restrictive as the congruency criteria applied to T-ANALYZE.

Step 8. The primary output displays the clusters in a sorted summary form where the largest clusters are presented first, and the transformation with the largest desired effect in each cluster is displayed as the cluster representative. The other data given are the compound identifiers, their activities, and the counts of the number of transformations within the cluster, which either increase or decrease or are equivalent in the activity. These counts are dependent upon the 'equivalence' value entered in step 5 and are provided as a quick way for the user to assess the worth of a cluster (as the Agreement and Z-score values do for T-ANALYZE). Within a cluster, the transformations are sorted by Diff, with the sort order dependent upon whether the desired direction

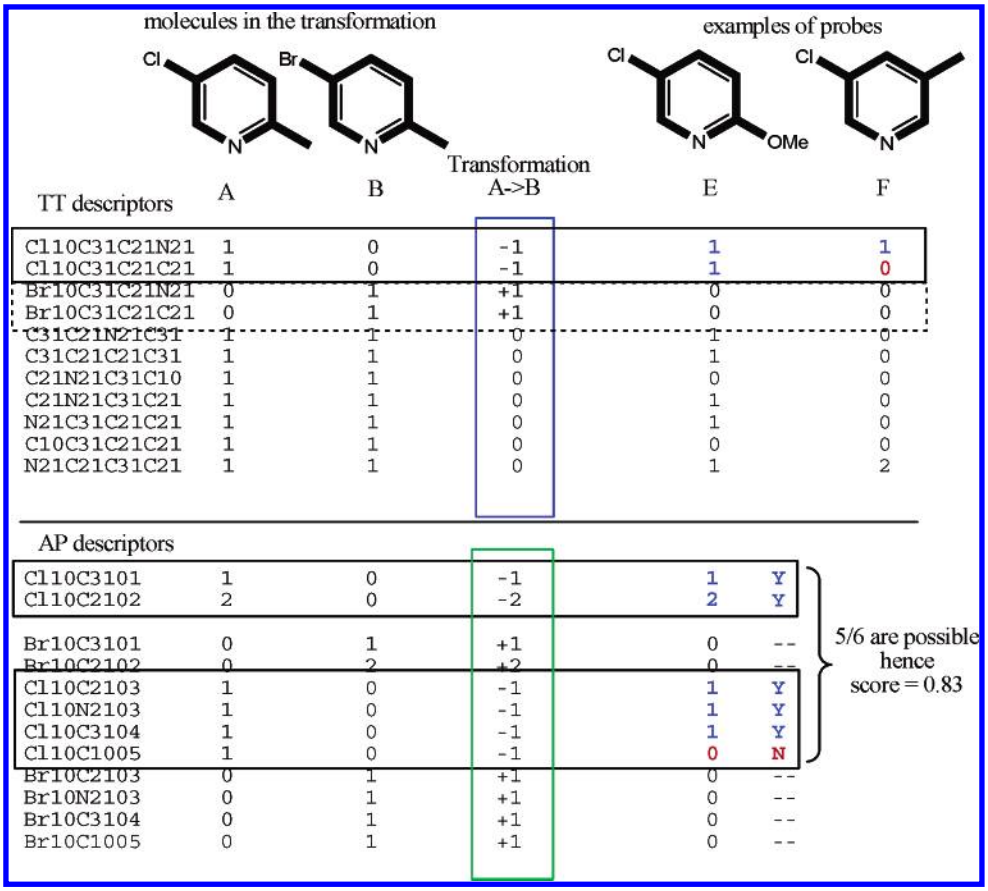


Figure 4. An illustration of how a transformation is judged to be applicable to a probe for the T-MORPH application. On the left is a pair of molecules which make up the stored transformation and their TT descriptors. The transformation vector is shown in the vertical blue box. On the right are two probes with their relevant TT descriptors. Probe E can have the transformation A→B applied to it while probe F cannot because it does not contain both descriptors which are required to be removed (those enclosed in the solid horizontal box). The dashed horizontal box encloses those descriptors of importance for the reverse transformation B→A. The applicability score is illustrated in the lower half of the panel. The AP descriptor vector is shown in the green box (other descriptors are present but are not listed for clarity), and the score is derived from the ratio of those AP descriptors which can be applied to the probe over the total number of negative AP descriptors in the vector.

entered in step 5 is 'increase' (greatest +ve Diff first) or 'decrease' (greatest −ve Diff first). Each transformation is shown in the direction A→B, where A is the molecule more like the probe and B is the molecule after the transformation is applied. This is the default ordering method, but an alternative means of sorting the transformations (either within a cluster or over the whole answer set) is available and can be thought of as a 'relevance score'.

This relevance score is derived from the comparison of the *negative* AP descriptor changes within the retrieved transformation, to the probe's AP descriptors and is illustrated in the lower half of Figure 4. The transformation A→B is applicable to probe E, hence one examines the negative AP descriptors in that transformation (the green box) and the existence of those AP descriptors in the probe E. The more AP descriptors from the transformations that can be applied to the probe, the more relevant the transformation is deemed to be. As one can see in Figure 4, 5 out of the 6 negative AP descriptors can be applied to probe E and so the relevance score is 5/6 (i.e. 0.83). The AP descriptors were chosen because they cover all the heavy atom to heavy atom connectivities and so encapsulate the long-range environment of any transformation. This method is suitable and useful for small numbers of answers, but with large transformation databases (>100 000 transformations) one can retrieve more than 1000 answers and so the clustering approach described above is preferred.

RESULTS

T-ANALYZE. We show three examples for T-ANALYZE.

1. D2. Dopamine agonists from ref 6. There are 116 molecules, most close analogues. The activity is $-\log(IC_{50})$. T-ANALYZE produced 264 clusters.

2. DHFR inhibition for the rat liver enzyme. There are 397 compounds. The activity is −log(Ki). These were downloaded as supplementary material from ref 7. T-ANALYZE produced 474 clusters.

3. ACE inhibitors. This data set has 114 molecules, some of which are stereoisomers. The activity is −log(Ki). The activities and structures are also from the supplementary material in ref 7. We included stereochemical information in the analysis of this set by using APC descriptors. T-ANALYZE produced 128 clusters.

Selected clusters for these examples are in Figures 5−7. The clusters are sorted by decreasing Z-score. For the D2 set (Figure 5), the largest difference in activity is made by a 3-Cl, but there is only one such transformation in the data set (cluster 1). The first cluster where there is more than

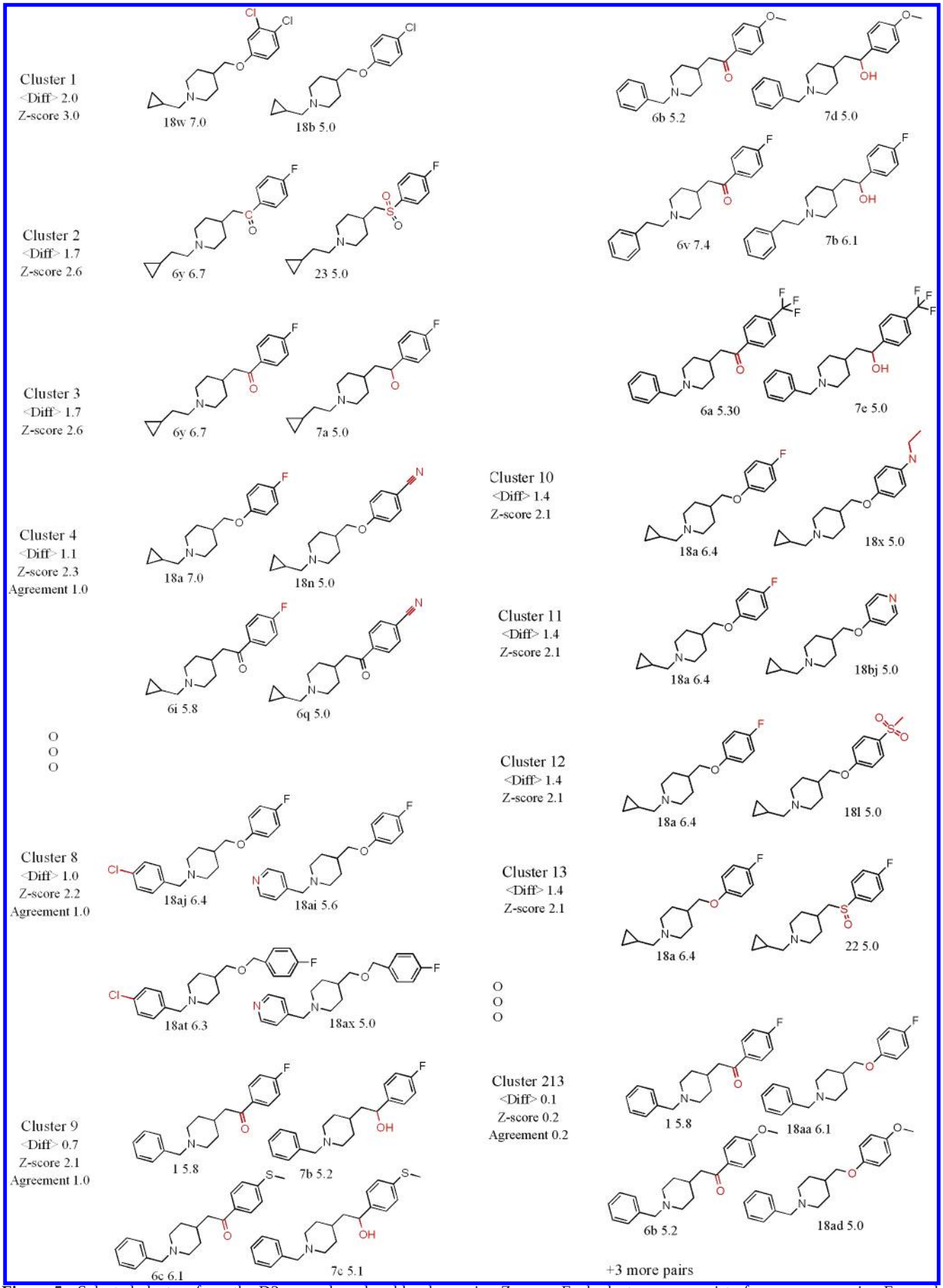
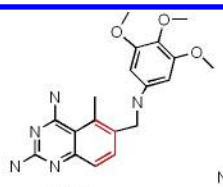
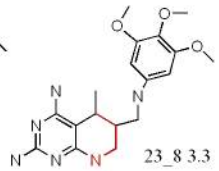


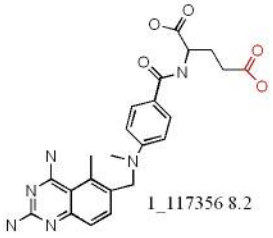
Figure 5. Selected clusters from the D2 example ordered by decreasing Z-score. Each cluster may consist of one or more pairs. For each molecule is listed the name of the molecule and its activity. The difference between the molecules in the pairs is highlighted in red. For clarity, where a transformation of a single atom involves a carbon, the symbol "C" is written for that carbon.

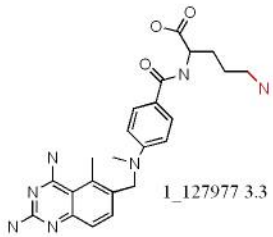
Cluster 1
<Diff> 5.2
Z-score 4.5


1 8.5

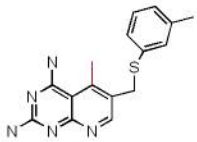

23_8 3.3

Cluster 2
<Diff> 4.9
Z-score 4.2

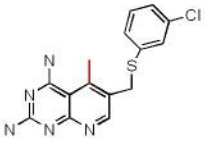

1_117356 8.2

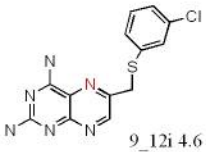

1_127977 3.3

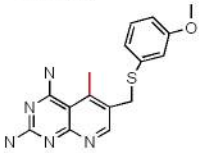
Cluster 3
<Diff> 1.7
Z-score 4.2
Agreement 1.0

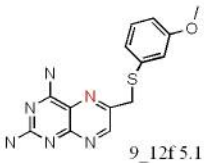

9_13d 6.5


9_12d 5.1


9_13i 6.7

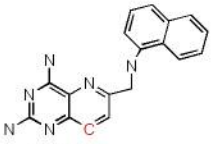

9_12i 4.6

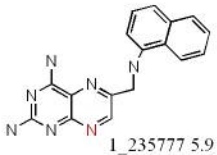

9_13f 6.2

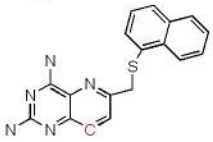

9_12f 5.1

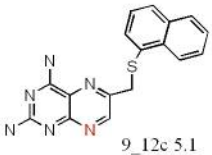
+7 additional pairs

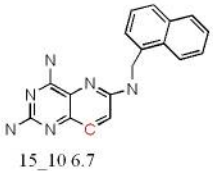
Cluster 4
<Diff> 1.7
Z-score 4.1
Agreement 1.0

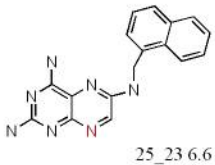

21_36 7.4


1_235777 5.9


21_34 6.8

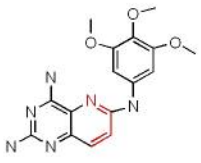

9_12c 5.1

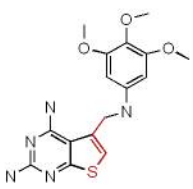

15_10 6.7

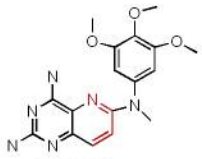

25_23 6.6

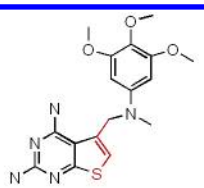
+ 6 more pairs

Cluster 5
<Diff> 3.6
Z-score 4.0
Agreement 1.0

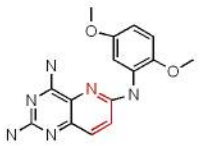

22_16 5.5

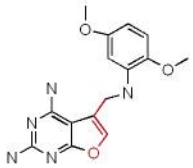

36_7a 3.3

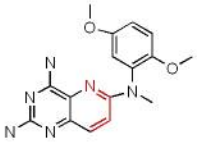

22_22 8.4

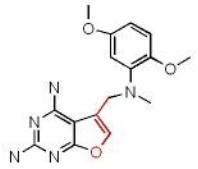

36_7c 3.3

Cluster 6
<Diff> 3.6
Z-score 4.0
Agreement 1.0

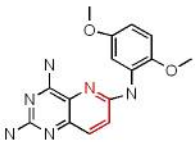

22_14 5.4

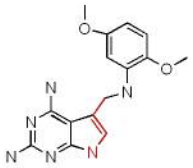

33_6 3.3

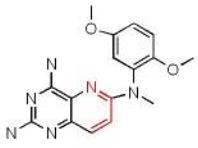

22_20 8.4

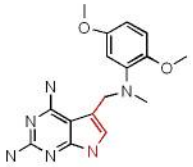

38_17 3.3

Cluster 7
<Diff> 3.4
Z-score 3.6
Agreement 1.0


22_14 5.4

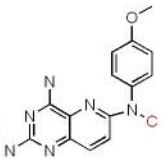

34_4 3.8

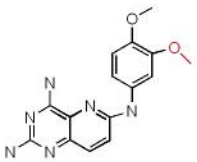

22_20 8.4

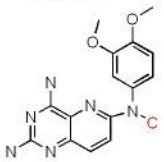

35_2 3.3

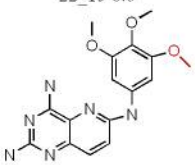
O
O
O

Cluster 9
<Diff> 3.1
Z-score 3.4
Agreement 1.0


22_19 8.2

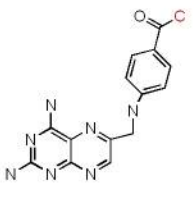

22_15 6.0

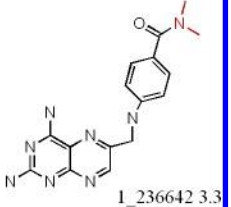

22_21 9.4


22_16 5.4

O
O
O

Cluster 11
<Diff> 3.8
Z-score 3.3


1_235791 7.1


1_236642 3.3

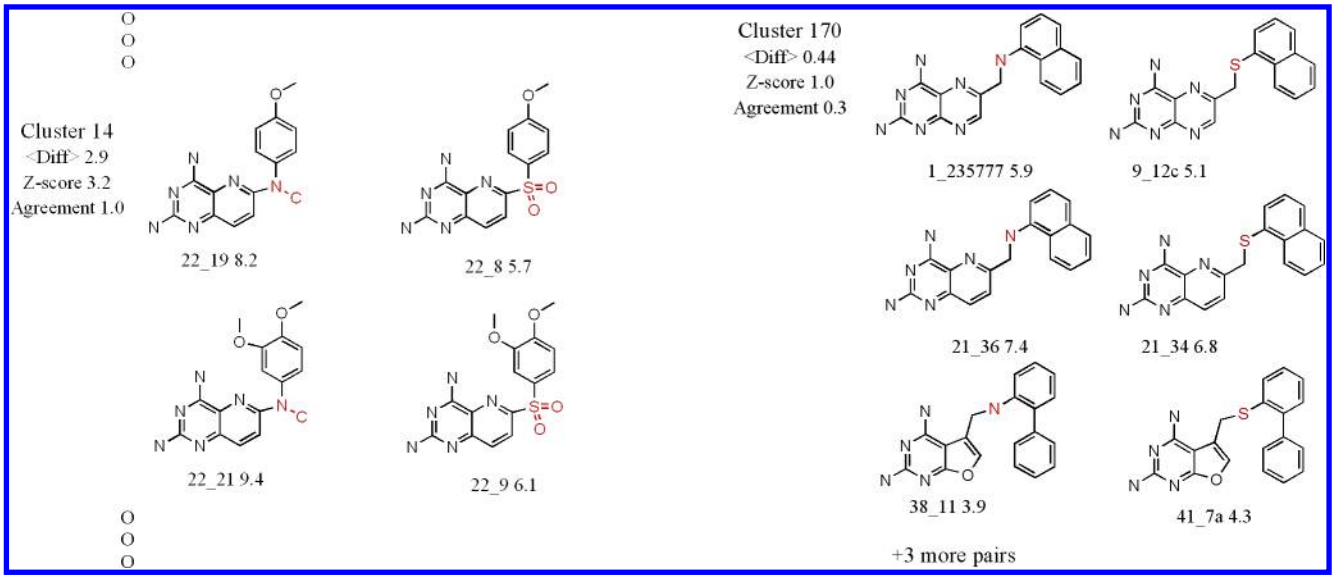


Figure 6. Selected clusters from the DHFR example ordered by decreasing Z-score. Conventions are as in Figure 5.

one transformation is cluster 4, where 4-F appears better than 4-cyano in two cases. Cluster 9 shows 5 examples where ketone is better than hydroxide. Note that cluster 3 is treated separately from cluster 9, because the molecules in those two clusters are sufficiently dissimilar by the criteria we use. One has to go fairly far down the list to find a large cluster where the Agreement of the cluster is $\ll 1.0$. Here we show cluster 213.

For DHFR (Figure 6) there are two transformations that have very large effects on activity, but there is only one example apiece (clusters 1 and 2). Cluster 3 is very large (10 transformations) and shows a consistent trend that changing the pyrazine portion of the pteridine to 4-methylpyridine increases the activity. Similarly cluster 4 (9 transformations) shows changing pyrazine to pyridine also increases activity. Clusters 5−7 show that changing the pyridine to another heterocycle decreases activity. Again, one has to go far down the list to find an example of a large cluster where the Agreement $\ll 1.0$. We show cluster 170.

For ACE (Figure 7), the largest change in activity is the change of phenylalanine for lysine (cluster 2). However, there is only one example. Cluster 1 shows that a change in stereochemistry in the proline ring is important. Not surprisingly cluster 3, which also affects the geometry of the proline, is consistent with this. Clusters 15 and 17 also show where stereochemistry is important. The nature of the Zn-binding portion of the molecule (−SH, −COO) is also, not surprisingly, important (clusters 4−7). Again we see a case where transformations appear similar to the eye (clusters 5 and 7), but the molecules are not similar enough that these transformations would be clustered together by our default criteria.

It is interesting to compare the transformations highlighted by T-ANALYZE to the results of standard descriptor-based QSARs. We ran QSAR fits of the activities against the AP (or APC) descriptor using trendvector,[8] a more general PLS method,[9] SVM (linear kernel),[10] and random_forest[11] and inspected the descriptor importances for each method. Different QSAR methods often disagree about what descriptors are the most important, so interpretation can be tricky. In the case of the D2 problem, all methods seem to agree. We interpret the most important AP descriptors to mean that the presence of 4-F-phenyl is associated with higher activity. Also the scaffold such as that seen in 6v (cluster 9, Figure 5), with a carbonyl and a piperidine separated by −CH2−CH2− from an aromatic group, is associated with high activity. On the other hand having a cyclopropyl on the piperidine is associated with low activity. In the case of DHFR, each QSAR method selects a different part of the molecule to emphasize. For instance, PLS notices that molecules with the 5-methylphenyl (as in cluster 3, Figure 6) is associated with high activity, while random_forest notices that the pteridine ring should be completely aromatic for high activity (as in cluster 1, Figure 6). For ACE most of the QSAR methods (using the APC descriptor) notice that molecules containing C(S-configuration)-CO-proline are the most active. SVM notices that a sulfhydryl 3 bonds from the proline is associated with high activity. While we do not wish to belabor the descriptor-based global QSAR, which is not the focus of this paper, we do want to point out that only a few local trends are discernible in the global QSAR, and some of the global trends are not seen in the local QSAR.

T-MORPH. To illustrate the application of T-MORPH to a probe we have used the D2 and DHFR data sets. Taking the full data set for each target we generated transformation databases, with 1240 transformations in the D2 and 6805 transformations in the DHFR database. For D2 we used 6v as the probe and for DHFR we used 1_235791. The probes are shown in Figure 8.

The search of the D2 database retrieved 263 transformations clustered into 32 clusters (many involving 6v). The largest cluster, cluster 1 (40 transformations), covers the transformation of the ketone into an ether linkage, the next cluster (39 transformations) describes the change of the piperidine substituent from phenethyl to benzyl, while cluster 3 (27 transformations) shows the effects of adding para substituents to an aryl ring at either end of the molecule. Cluster 4 (20 transformations) is like cluster 2 but deals with the piperidine substituent changing from cyclopropyl ethyl to cyclopropyl methyl, and cluster 5 (19 transformations) deals with the change of a phenyl ring into a biphenyl or naphthyl substituent. Taking cluster 6 as an example in Figure 9, this has 17 transformations, 2 of which involve

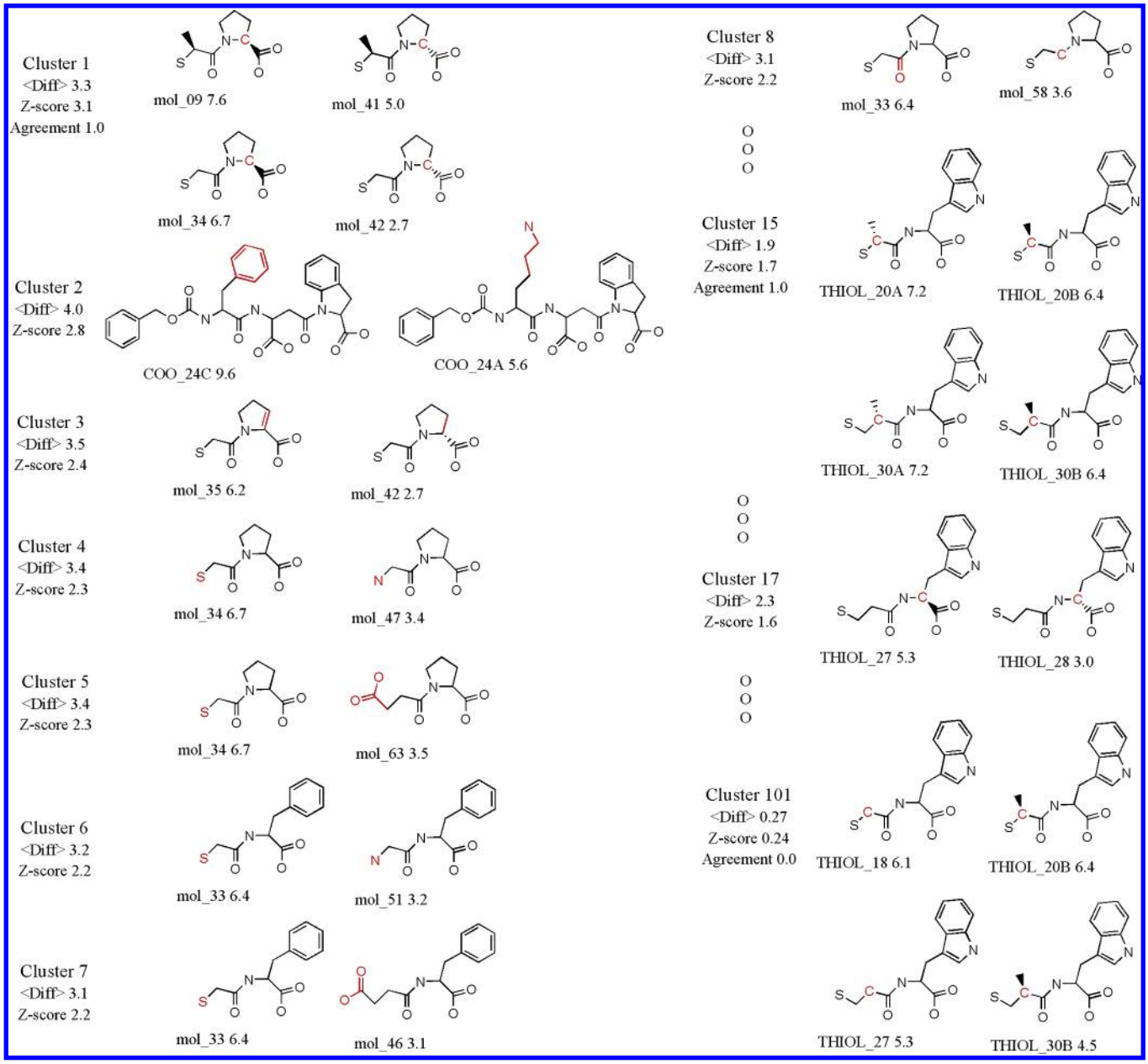


Figure 7. Selected clusters from the ACE example ordered by decreasing Z-score. Each cluster may consist of one or more pairs. Conventions are as in Figure 5. Stereochemistry is indicated where it is important for the transformation.

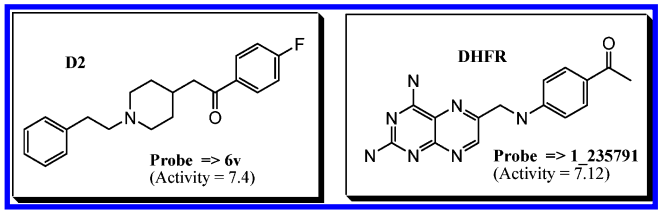


Figure 8. The probe molecules used to search the transformation database created from the D2 and DHFR data sets.

the probe 6v and all of which deal with the reduction of the ketone to an alcohol although other transformations (such as the chain extension of the piperidine substituent in example 15 or the swap of the para-fluorine for a methoxy substituent in example 9) are also involved. In the real life use of T-MORPH this cluster would help one recommend that 6v *not* be reduced as this would lead to lower activity, a result which is borne out by example 14 in the cluster. This cluster is highlighted in order to compare the T-MORPH output with the T-ANALYZE analysis of the D2 set. Overall there are many fewer clusters to examine in the T-MORPH output, and so only the highly relevant information is provided to the chemist. Indeed all 5 transformations listed in the T-ANALYZE cluster are also found in the T-MORPH cluster. In addition there are a further 12 transformations in the T-MORPH output which are closely related to the key reduction transformation; however, these extra transformations also involve other changes which may confound the effect of the reduction on activity. This clustering of related but not exactly the same transformation also occurs, to a similar extent, in the larger clusters (numbers 1−5) which were not described in detail. It is debatable whether these extra transformations in T-MORPH compared to T-ANALYZE are helpful or confusing, but we feel that this question is entirely context or user dependent and so both methods are provided for use.

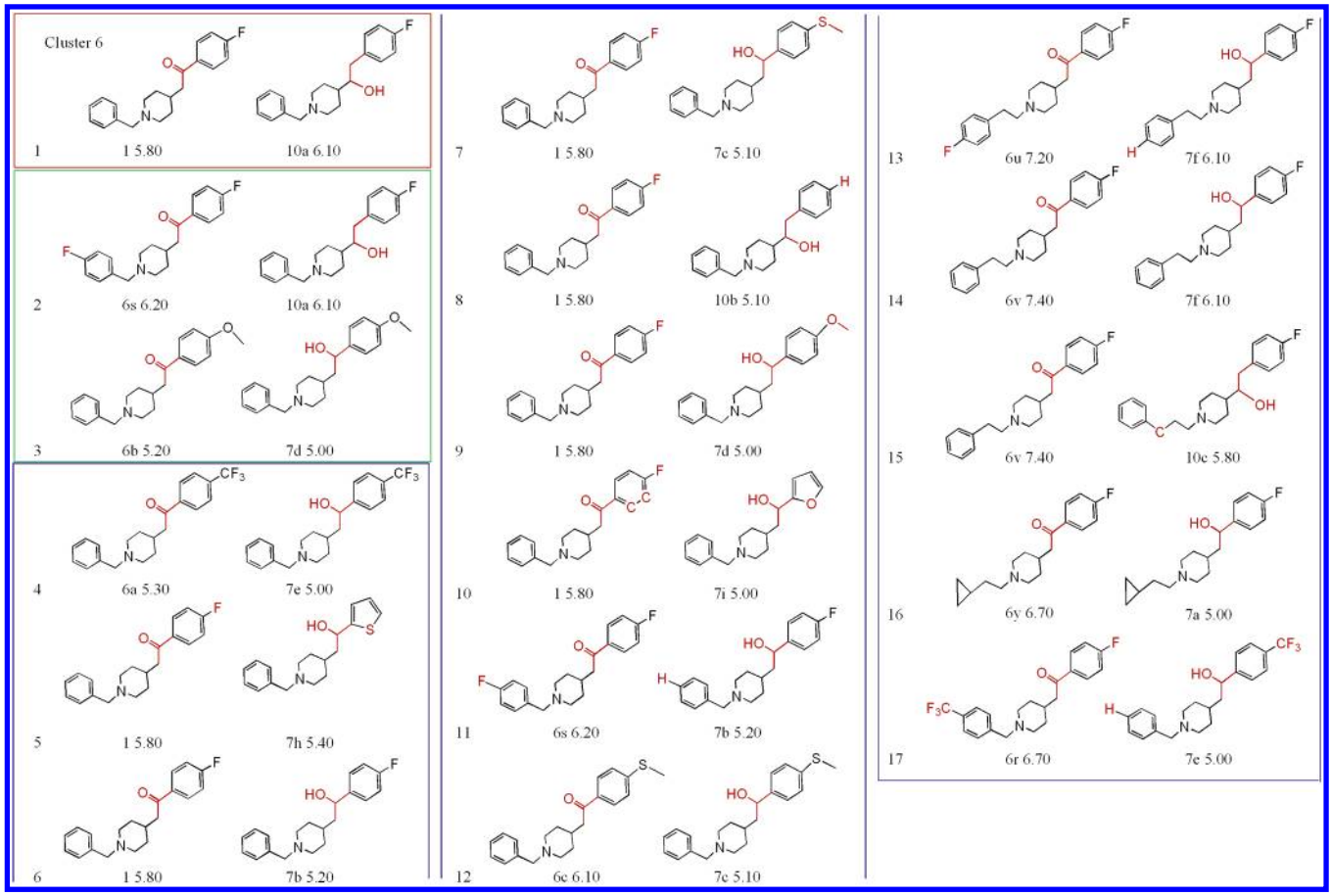


Figure 9. Cluster 6 of transformations which are retrieved, from the search of the D2 database, using the probe 6v shown in Figure 8. The transformations are displayed with their order number in the cluster, the molecule names, and activities. The difference between each pair of molecules is identified by the red atoms. Those transformations which are desirable (i.e. affinity increasing) are enclosed in the red box, those that would be considered as equivalent are in the green box and those where the activity change is undesirable are in the blue box. This cluster deals almost solely with the reduction of the aryl ketone to an alcohol.

The search of the DHFR database retrieved 166 transformations clustered into 20 clusters (many involving 1__235791). Cluster 1 contains 37 transformations, 14 of which involve 1__235791 and which detail the change of the upper nitrogen of the pyrazine part of the pteridine system into either a C-methyl or show its removal altogether to form a five-membered ring. These changes indicate that the replacement of that nitrogen with a C-methyl is favorable for activity, while the ring contraction is not. Cluster 2 (Figure 10) contains 25 transformations, of which 13 involve the probe 1__235791, and describe changes to the aryl ketone (e.g. changing it to various −Cl or −OMe or −H substitution patterns). However, these pendant aryl ring transformations also include some compounds where the lower pyrazine nitrogen is also converted into a carbon. The changes to the ketone appear to be detrimental to activity, but one would also want to determine the effects of only the pyrazine nitrogen to carbon transformation. For this one would have to go down to cluster 6 (11 transformations, only 1 involving the probe), which contains this transformation, and one finds that all increase activity.

Therefore, again, if this were a real project, one would suggest this pyrazine N to C change for the probe 1__1235791 and also suggest that the aryl ketone not be changed. Within the real data (e.g. number 9 within cluster 6, 1__1235791 to 21__40), one can see that the N to C change did increase the affinity of 1__1235791, and example 25 in cluster 2 (changing 1__1235791 to 1__235776) or the related example 22 (changing 21__40 to 21__4) neatly demonstrates the affinity reducing effect of removing the methyl ketone.

DISCUSSION

We have presented a methodology for mimicking the process by which chemists perceive local QSAR but making the process automatic and more systematic. It is important to note that the methodology does not involve building any type of statistical model but merely organizes existing data. The T-ANALYZE methodology is a great practical help to people trying to understand the local QSAR in a large unfamiliar data set. In particular we note how many clusters, i.e., types of transformations, appear in fairly small data sets such as the ones we show here and have an appreciation of how unlikely it is that a "manual" inspection of that data would find more than a small minority of them.

T-ANALYZE is a useful complement to methods that generate global descriptor-based QSARs. We have seen in all our cases that descriptor-based QSARs are not easily interpreted in terms of local effects, and the local QSAR does not necessarily show global trends.

Even when the local QSARs are enumerated, the user is still faced with having to go through all of the clusters to see which transformations would be applicable to a molecule/series of current interest to improve the activity, and so we

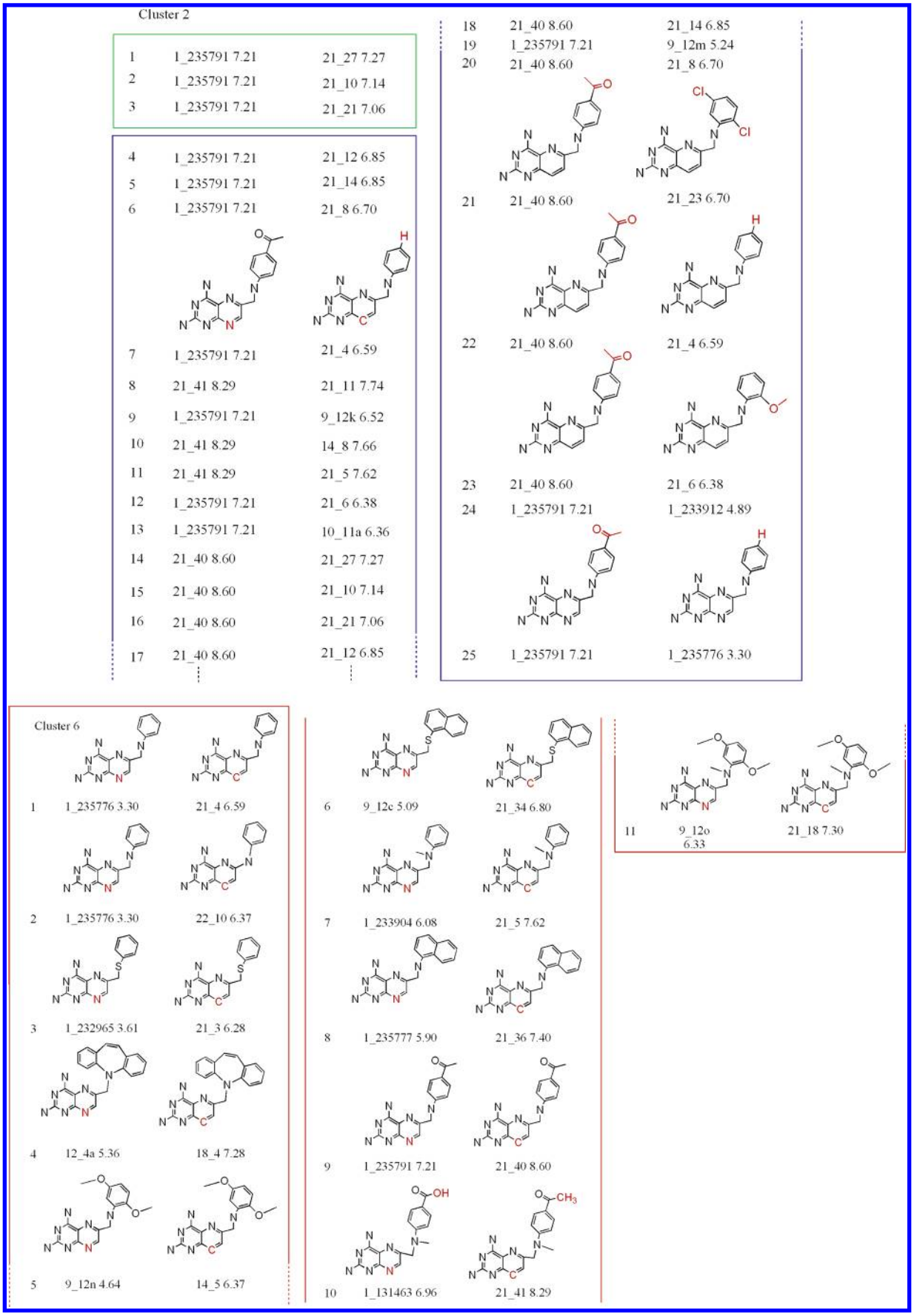


Figure 10. Clusters 2 and 6 of transformations which are retrieved, from the search of the DHFR database, using the probe 1_235791 shown in Figure 8. The conventions are the same as Figure 9. Some transformations are listed in text form for space reasons except for those transformations explicitly mentioned in the main text. In cluster 2 the transformations detail changes to the aryl ketone which appear to be detrimental to activity but also include some changes to the lower pyrazine nitrogen. Cluster 6 deals almost solely with changes to the lower pyrazine nitrogen and show that the change to the carbon is always beneficial to activity.

also present a second methodology T-MORPH to aid that step. We note in this context that Lewis[12] recently published a method of applying a QSAR to suggest improvements in a molecule, but that method uses (descriptor-based) global rather than local QSAR. Suggestions based on statistical models relating descriptors to activity tend to be opaque to the average medicinal chemist. In contrast our T-MORPH application displays explicit compounds that illustrate the suggested change. Lewis has also raised the issue that the automatic application of some descriptor-based QSARs to a molecule can lead to ridiculous modifications if there are no inhibitory feedback loops in place, i.e., if increasing molecular weight is good, the molecule tends to grow without limit, or if adding a fluorine to a phenyl group is beneficial, then adding 5 fluorines to the molecule is perceived as 5 times as good. One can see from the simple example of Figure 4 that the use of the transformation vector methodology does not have this problem; as soon as the suggested modification is made to a probe the set of TT descriptors change (note how much change in the TT descriptors there is between probes E and F). This makes the application of the same transformation multiple times impossible.

There is a lot of subjectivity involved in what to consider an interesting transformation and what test of congruency among transformations should be used. Our results for either application depend, of course, on the descriptors we use and the cutoffs. First, there is no substructure descriptor that will completely describe chemical space to everyone's satisfaction (though in our experience the AP and TT descriptors are very good), and there will always be disputes about whether the descriptor has grouped compounds correctly. In the case of the T-ANALYZE application, because of the N^4 nature of the compute time, there is high motivation to prune out a large fraction of transformations before comparing transformations and also to prune out pairs of transformations, hence the use of fairly stringent filters. However, some filters can be too stringent under some circumstances and eliminate some interesting congruities. For instance, one time-saving filter in the MCS approach is the assumption that two transformations A→B and C→D are potentially congruent when A, B, C, and D are related molecules, that is, when the context of the transformation is similar. The philosophy behind this is that we do not expect adding a Cl to a benzodiazepine to have the same effect as adding a Cl to a beta-carboline, and we do not necessarily expect a Cl at the 7-position to have the same effect at the 3-position. Thus, those types of changes are seldom clustered under our current parameters. Similar changes to dissimilar series will still be found, just not in the same cluster. In contrast, in the T-MORPH application, where the computational requirements are smaller, we allow similar changes in dissimilar molecules to be clustered, and in some cases it is useful to see these together. Adding a fluorine to an aromatic ring to inhibit metabolism is an example of a transformation that is applicable to many chemical types. Finding substituent effects on groups which bind in the same binding pocket while being on different cores is another.

On the other hand, it should be noted that our criteria allow for the clustering of transformations that some might feel do not belong together. It is debatable whether to cluster transformations that involve more than one position on the molecules. For instance in both the T-ANALYZE and T-MORPH applications, if the molecule is nearly symmetric the descriptors may not be able to distinguish a Cl→Br transformation at one end of the molecule from a Cl→Br transformation at the other. However, clustering transformations that do not appear the same by eye is relatively rare in T-ANALYZE. In the T-MORPH application, where there is no MCS filter, we allow for more congruences and more problematic transformations can appear. Some specific quirks are that a ring expansion (such as from a pyrrolidine to a piperidine ring) or a chain extension (such as from a butyl to pentyl chain) involve only increases in TT descriptors and no removal of TT descriptors. Hence if a transformation does not have negative TT descriptors there is nothing required to be taken away from a probe, and so these transformations are applicable to every probe. While one could remove this sort of transformation from the database it would also remove some valuable information, and so such transformations have been retained. We consider current cutoffs, specific to each application, a reasonable compromise between clustering interesting transformations and not including too many uninteresting ones. The cutoffs are under user control, however, and can be tuned for a particular problem.

The N^4 nature of the T-ANALYZE application means that problems do not scale particularly well. While data sets of a few hundred compounds take a few seconds to process on an IBM PWR4 processor, many data sets from pharmaceutical companies consist of thousands or tens of thousands of compounds. Serial processing of the latter could take months. We have been able to parallelize parts of the method so that the job is divided among up to 80 processors. That way we have been able to address a problem with 20 000 compounds in 36 h elapsed time. The T-MORPH approach has not required, as yet, to be parallelized as the production of a transformation database takes less than 2 h for a data set of 6371 compounds giving 121 743 transformations. The search of such a database is complete in less than 3 min on an R12K SGI Octane, with the clustering of the results taking an extra 12 min (wall clock time) for a result set of approximately 1800 transformations. These timings are obviously dependent upon the nature of the probe but are reasonably representative.

ACKNOWLEDGMENT

The tools underlying T-ANALYZE and T-MORPH fall under the in-house modeling system MIX, and the authors thank all those involved in MIX development.

Note Added after ASAP Publication. This article was released ASAP on November 3, 2005 with a few minor text errors and an incorrect definition of ACE inhibitors. The correct version was posted on November 8, 2005.

REFERENCES AND NOTES

(1) Carhart, R. E.; Smith, D. H.; Ventkataraghavan, R. Atom pairs as molecular features in structure−activity studies: definition and application. *J. Chem. Inf. Comput. Sci*. **1985**, *25*, 64−73.

(2) Nilakantan, R.; Bauman, N.; Dixon, J. S.; Venkataraghavan, R. Topological torsion: a new molecular descriptor for SAR applications. comparison with other descriptors. *J. Chem. Inf. Comput. Sci.* **1987**, *27*, 82−85.

(3) Sheridan, R. P.; Miller, M. D. A method for visualizing recurrent topological substructures in sets of active molecules. *J. Chem. Inf. Comput. Sci*. **1998**, *38*, 915−924.

(4) Sheridan, R. P. The most common chemical replacements in drug-like compounds. *J. Chem. Inf. Comput. Sci*. **2002**, *42*, 103−108.
(5) Butina, D. Unsupervised database clustering based on Daylight's fingerprint and Tanimoto similarity: a fast and automated way to cluster small and large data sets. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 747−750.
(6) Gilligan, P. J.; Cain, G. A.; Christos, T. E.; Cook, L.; Drummond, S.; Johnson, A. L.; Kergaye, A. A.; McElroy, J. F.; Rohrbach, K. W.; Schmidt, W. K.; Tam, S. W. J. Novel piperidine sigma receptor ligands as potential antipsychotic drugs. *J. Med. Chem.* **1992**, *35*, 4344−4361.
(7) Sutherland, J. J.; O'Brien, L. A.; Weaver, D. F. A comparison of methods for modeling quantitative structure−activity relationships. *J. Med. Chem.* **2004**, *47*, 5541−5554.
(8) Sheridan, R. P.; Nachbar, R. B.; Bush, B. L. Extending the trend vector: the trend matrix and sample-based partial least squares. *J. Comput.-Aided Mol. Des*. **1994**, *8*, 323−340.
(9) pls.pcr Package in R by Ron Wehrenshttp://cran.r-project.org/src/contrib/Descriptions/pls.pcr.html.
(10) LIBSVM by Chih-Chung Chang and Chih-Jen Lin http://www.csie.ntu.edu.tw/∼cjlin/libsvm/.
(11) Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J. C.; Sheridan, R. P.; Feuston, B. P. Random forest: a classification and regression tool for compound classification and QSAR modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1947−1958.
(12) Lewis, R. A. A general method for exploiting QSAR models in lead optimization. *J. Med. Chem.* **2005**, *48*, 1638−1648.

CI0503208