Cell-Based Analysis of High Throughput Screening Data for Drug Discovery

Raymond L.H. Lam

Biomedical Data Sciences
GlaxoSmithKline Inc
Mississauga, Ontario L5N 6L4
Canada

William J. Welch

Dept of Statistics & Actuarial Science
University of Waterloo
Waterloo, Ontario N2L 3G1
Canada

S. Stanley Young

Statistical Research Unit
GlaxoSmithKline Inc
Research Triangle Park,
North Carolina 27709-3398
USA

Cell-Based Analysis of High Throughput Screening Data for Drug Discovery

Abstract

One of the first steps of drug discovery is finding chemical compounds with some activity for the chosen biological target. This search is often done by randomly screening very large numbers of compounds, thousands to hundreds of thousands. In contrast, we propose a cell-based analysis method that guides selection of compounds for screening. Starting with a screen of a relatively small subset of compounds, we identify small regions in a high-dimensional descriptor space that have a high proportion of active compounds. We use these regions to score and prioritize untested compounds for further screening. Our method is capable of finding multiple active regions and increasing the rate of finding active compounds many times over random screening.

KEY WORDS: Uniform coverage designs, High-dimensional space, Multiple mechanisms, Recursive partitioning, Classification, Scoring.

1. Background

In screening for drug discovery, thousands to hundreds of thousands of chemical compounds are screened in the hope of discovering biologically active compounds. The evaluation of a single compound can cost from a few cents to several dollars depending upon the complexity of the assay. At the next stage of drug development, the active compounds or “hits” found by screening are typically modified atom-by-atom to improve activity and other important characteristics, such as tissue distribution, plasma half-life, toxicity, etc. The aim of the initial screen, then, is to find active compounds of several structurally different chemical classes, to provide a variety of starting points for subsequent optimization.

In addition to finding active compounds among those screened, it would be very useful to know how to find additional active compounds without having to screen each compound individually. We might initially screen part of a collection and use these data to predict which compounds in the remainder of the collection are likely to be active. Several cycles of screening are expected to be more efficient than screening all the compounds in a large collection (Jones-Hertzog et al. 2000). To do this we need to analyze the initial high throughput screening (HTS) data to find association rules linking biological activity (response variable) to specific values of the compound descriptors (explanatory variables).

The first step in the process of determining features of compounds that are important for biological activity is describing the molecules in a relevant, quantitative manner. A drug-like molecule is a small three-dimensional object that is often drawn as a two-dimensional structure. This two dimensional graph is subject to mathematical analysis and can give rise to numerical descriptors to characterize the molecule. Molecular weight is one such descriptor. There are many more. Ideally, the descriptors will contain relevant information and be few in number so that the subsequent analysis will not be too complex. To exemplify our methods we use a system of 67 BCUT descriptors (Section 2.3).

The relationship between descriptors and activity is extremely complex for HTS screening data, and there are several challenges in statistical modeling. First, the potent compounds of different chemical classes may be acting in different ways. Different mechanisms might require different sets of descriptors within particular regions (of the descriptor space) to operate, and a single mathematical model is unlikely to work well for all mechanisms. Also, activity may be high for only very localized regions. Second, even though a design or screen may include thousands of compounds, it will usually have relatively few active compounds. The scarcity of active compounds makes identifying these small regions difficult. Third, there are many descriptors (i.e., curse of dimensionality) and they are often highly correlated. This is the case for BCUT numbers. Fourth, many HTS data sets have substantial measurement error. Because of some or all of these complexities, common statistical analysis methods such as linear regression models, generalized additive models, and neural nets are ineffective in handling these analysis problems (Young and Hawkins, 1998) and tend to give low accuracy in classifying molecules as active.

The rest of the paper is organized as follows. In Section 2 we describe two motivating data sets. Section 3 expands on the difficulties that current methods face with complex structure-activity relationships. In Section 4 we present a cell-based analysis method that overcomes these problems. It divides a high-dimensional (descriptor) space into many small, low-dimensional cells, scores cells according to the activities of their compounds, and uses the scores to prioritize further compounds for screening. This analysis method is highly related to the uniform cell coverage approach described by Lam et al. (2001) for selecting molecules for screening. Thus, the earlier work and the current article together provide an overall strategy for design and analysis of HTS data. In Section 5 we evaluate our analysis approach on the two data sets and show that it can improve prediction accuracy compared with recursive partitioning (trees), one of the few successful methods for HTS structure-activity data. Finally, Section 6 makes some conclusions and discusses further work.

2. Motivating Applications

The new method described here can be applied to both continuous and discrete responses. For illustration, a data set with continuous activity outcome (Core98) and a data set with binary activity outcome (NCI) are included.

2.1. Core98 Molecular Data (Continuous Response)

Core98 is a chemical data set from the GlaxoSmithKline collection. Activity is available for 23,056 compounds. The response is % Inhibition for a given biological target and theoretically should range from 0 to 100%, with more potent compounds having higher scores. Biological and assay variations can give rise to observations outside the 0-100% range. Typically, only about 0.5% to 2% of screened compounds are rated as potent.

2.2. NCI Molecular Data (Binary Response)

An AIDS antiviral screen chemical database can be obtained from the National Cancer Institute (NCI) web site http://dtp.nci.nih.gov/docs/aids/aids_data.html. It provides screening results and chemical structural data on compounds. When we downloaded the database in May 1999, there were about 32,000 compounds. GlaxoSmithKline computational chemists generated BCUT numerical molecular descriptors (see Section 2.3) for these compounds. However, due to poor structural representation and samples that contain unusual chemical substances that would normally not be considered drug candidates, some BCUT descriptors could not be computed for some compounds. These compounds were removed, leaving about 30,000 compounds with computed descriptors.

Unlike the Core98 data where the response is continuous, the NCI compounds are classified as moderately active, confirmed active, or inactive. We combine the first two categories into a single active class to give binary response data, as there are only 608 (roughly 2%) active compounds.

2.3. Descriptor Variables

For both data sets we use BCUT descriptors based on the work of Burden (1989) to describe the compounds. The BCUT descriptors are eigenvalues from connectivity matrices derived from the molecular graph. The square connectivity matrix for a compound has a diagonal element for each heavy (non-hydrogen) atom. The diagonal values are atomic properties such as size, atomic number, charge, etc. Off diagonal elements measure the degree of connectivity between two heavy atoms. Since eigenvalues are matrix invariants, these numbers measure properties of the molecular graph and hence the molecule.

When we first started this research, only six BCUT descriptors were available to us. They were used in development of a uniform coverage design method (Lam et al., 2001). Subsequently, GlaxoSmithKline computational chemists also provided a larger set of 67 descriptors for the motivating applications. The larger set was suggested by Pearlman and Smith (1998). We found that the 67 BCUT descriptors are highly correlated in the two data sets. A reason for the high correlations is that scientists often devise descriptors that measure the same general property of a compound.

While our software for the cell-based analysis method can handle 67 descriptors, the computational time is much larger. For example, it takes roughly 100 hours versus 5 minutes for 67 versus 6 descriptors. Thus, we primarily use the smaller set in this paper. The current software (written in SAS code) was aimed at testing the new methods and did not focus on efficiency in dealing with large data sets with many variables. We plan to implement the cell-based analysis algorithm using C++ code, which should run hundreds of times faster than the current software. Whether the larger set of descriptors has substantially more predictive power is a question of some interest to the computational chemists, however, and we make some comparisons in Section 5.

2.4. Dividing Data into Training and Validation Sets

For the purpose of demonstrating the validity of the new methods, we divide each of the original data sets into training and validation sets. We use the training data (treated as screened compounds) to build models (i.e., find active regions) and the validation data (treated as unscreened compounds) to evaluate prediction accuracy (i.e. verify if the activity in these regions remains high). The validation set gives a more unbiased evaluation of the statistical method than the training set. In real applications we would use all the assayed compounds to find active regions, as more data increases the prediction power.

There are 608 active compounds (roughly 2%) in the NCI data set. This population or random hit rate of 2% gives us a benchmark for the performance of our analysis method. If an analysis method gives hit rates (proportion of active compounds amongst those selected) in the validation set many times higher than the random hit rate, then it performs well

For the Core98 compounds, the activity response variable is on a continuous scale. The mean, standard deviation, and median of the measured activities are 7.8, 8.9, and 5.9%, respectively. We refer to the mean activity as the population or random activity value. As well as analyzing the data on this scale we can also classify the compounds with the top 1% of measured activities as active. This 1% random hit rate corresponds to 34.8% inhibition on the continuous scale. The population mean activity of 7.8% inhibition (continuous response) or the population active hit rate of 1% (binary response) again provide benchmarks for the analysis methodology.

We will use relatively small training sets, as one of our goals is to predict from a small screening design. The training molecules will be selected either using the Lam et al. (2001) uniform-coverage design algorithm or at random. With a 1-2% hit rate, a sample size of 4096 compounds gives roughly 40-80 active compounds, which should be sufficient to build a sound prediction model. (A sample size of 4096 is a convenient number for the design algorithm.) Table 1 shows the expected division of active compounds between the training and validation sets for the NCI data and for the Core98 data (binary response).

Table 1. Expected Distribution of Active Compounds Between a Training Set of 4096 Compounds and a Validation Set of the Remaining Compounds For Random Designs

	All data	Training set	Validation set
Data set	# actives / # compounds	# actives / # compounds	# actives / # compounds
NCI	608 / 29 812	84 / 4 096	524 / 25 716
Core98	231 / 23 056	41 / 4 096	190 / 18 960

3. Existing Methods

Here we describe two statistical analysis methods commonly used for analyzing chemical data sets.

3.1. Cluster Significance Analysis

Cluster significance analysis (CSA), introduced by McFarland and Gans (1986), aims to find embedded regions of activity in a high dimensional chemical space. CSA considers every subspace that can be formed by the predictors, from all one-dimensional subspaces up to the space of all predictors. A subspace is simply a subset of the descriptor variables, ignoring the rest. For each subspace, CSA computes the average distance between the active compounds and compares the average to the distribution of average distance for an equal number of compounds randomly selected from all compounds (active or inactive). If the actives are clustered tightly, as measured by a randomization significance test, this is evidence that the descriptors forming the subspace and the regions where the actives are clustered are important for activity.

A synthetic data set is instructive of the method and the potential problems. CSA tacitly assumes that there is only one class of active compounds forming one cluster in one or more subspaces. Suppose, however, that there are two mechanisms operating. (In practice, we would not necessarily know which mechanism is causing activity, nor even how many there are.) Mechanism M1 active compounds require that the descriptor molecular weight is between 400 and 500 and that the melting point is between 160 and 205 degrees C. These active compounds are denoted by squares in Figure 1(a). Mechanism M2 active compounds require that the descriptor LogP (the octanol and water partition coefficient) is in the range 3.0-4.0; they are shown by circles in Figure 1(a). Dots represent inactive compounds. Because molecular weight and melting point are unimportant for mechanism M2, the circles are spread throughout the subspace, making it difficult to detect clustering of the actives. Similarly, if we look at a subspace that includes LogP, as in Figure 1(b), the M1 actives are spread across the LogP dimension. Even in this somewhat simple situation, the CSA algorithm could have trouble. Similarly, if there are two or more active regions in a single subspace, in principle, a single measure of clustering might not detect them.

3.2. Recursive Partitioning Approach

The analysis of multi-mechanism data is difficult, and many statistical methods are not expected to be successful. Recursive partitioning (RP), Hawkins and Kass (1982) and Breiman et al. (1984), is one method that has been successful with multiple mechanisms arising in drug-screening data (Hawkins et al. 1997, Young and Hawkins 1998, Rusinko et al. 1999, and Jones-Hertzog et al. 2000). RP selects a descriptor to partition the data into two or more groups or nodes that are more homogeneous. Each daughter node is partitioned in turn until the nodes are judged homogeneous or some minimum sample size is reached. This separation of the data into smaller groups can, at least in principle, isolate the active compounds due to a single mechanism.

As successful as RP has been for the analysis of HTS data sets, there are a number of possible problems. These problems are at least partially due to particular implementations of RP in existing software products, rather than the overall concept. First, RP selects one descriptor at a time to split the data set, but a single descriptor may not provide adequate information for the splitting process. In addition, when the descriptors are highly correlated, selecting one will likely lead to never selecting several others. It is important to keep the following observation in mind: two compounds must have fairly close values of all critical descriptors for similar biological activity (McFarland and Gans, 1986) when there is a single mechanism. This means that partitions have to be narrow, and in several dimensions simultaneously, if all molecules from a partition are to have similar activity. The second problem relates to multiple

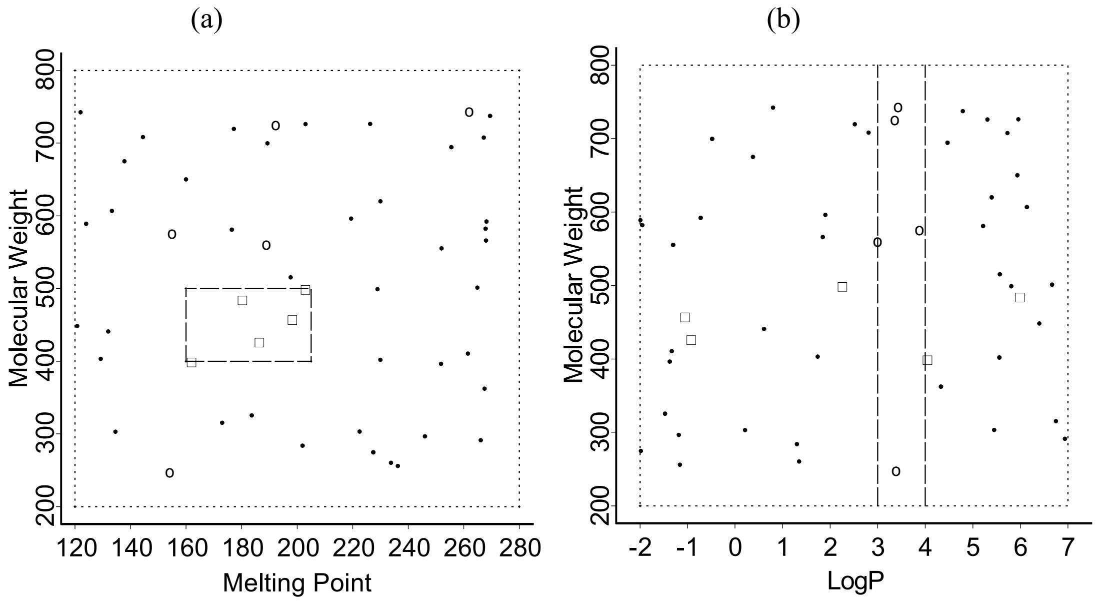


Figure 1. Distributions of Active Compounds from Two Mechanisms. Squares and circles represent compounds active via Mechanisms 1 and 2, respectively, while dots are inactive compounds. Active regions corresponding to these mechanisms are shown by dashed lines: (a) locations of compounds in the subspace formed by Molecular Weight and Melting Point, and (b) locations of compounds in the subspace formed by Molecular Weight and LogP.

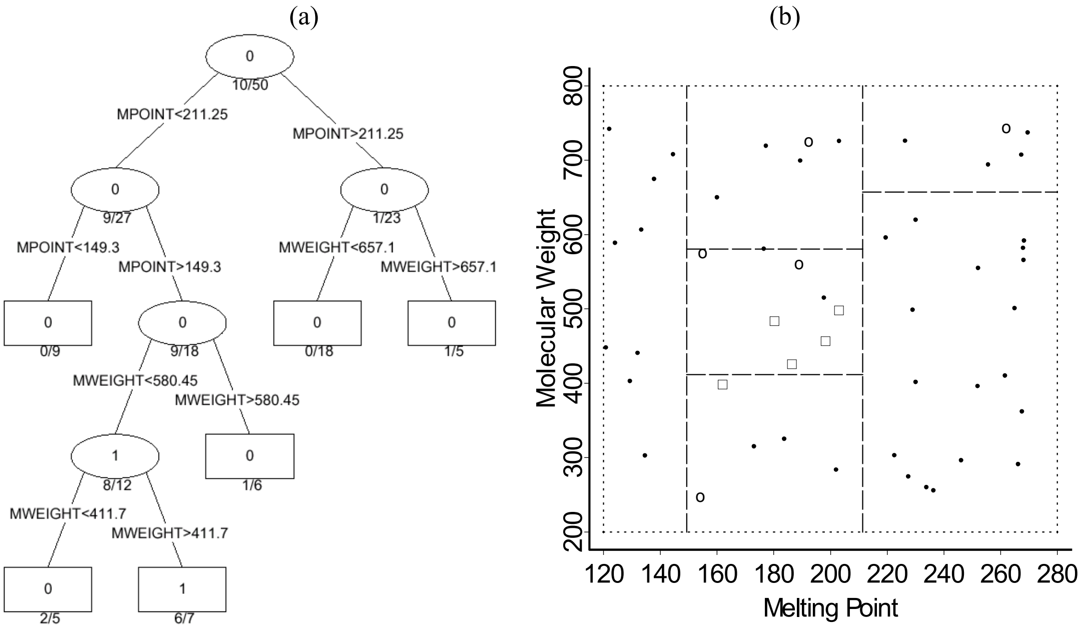


Figure 2. Recursive Partitioning of Two-Mechanism Data. Recursive partitioning (S-Plus tree with default settings, e.g., minimum node size of 5) is used to split the data illustrated in Figure 1. (a) Nodes in the tree are classified as active and inactive and are labeled by 1 and 0, respectively. Terminal nodes are represented by rectangles. Under each node, the hit rate is printed. (b) The corresponding partitions are displayed.

mechanisms when active compounds from these mechanisms cannot be easily separated. The two-mechanism data shown in Figure 1 illustrate the problem. Figure 2(a) gives a tree, generated by recursive partitioning. Because logP is never chosen as a partitioning variable, the logP subinterval containing the Mechanism 2 active compounds is not identified. The tree partitions are displayed in Figure 2(b); RP incorrectly splits the subspace formed by Molecular Weight and Melting Point into six regions. Here, partitioning one variable at a time is ineffective in dealing with multiple mechanisms. The third problem relates to the use of binary splits in many implementations. Problems can result if the activity pattern is inactive-active-inactive for a descriptor variable. With a single cut point, actives will be combined with inactives, possibly leading to the variable not being selected.

4. Cell-Based Analysis

For convenience, we refer to a small region of a *d*-dimensional (sub)space as a *d*-dimensional cell. For example, a 2-D cell is a region of a 2-D space.

We introduce a cell-based analysis method that first identifies small regions (cells) with several active compounds in low-dimensional subspaces (projections) of a high-dimensional descriptor space and then uses the information on these cells to score new compounds and prioritize them for testing. The cell-based analysis algorithm involves five stages.

1. Divide the high-dimensional space into many tiny cells (Section 4.1).
2. Make a preliminary identification of good cells: those cells with several active compounds (Section 4.2). Cells with too few active compounds are removed as there is not enough evidence to achieve statistical significance.
3. Derive ranking scores for the good cells (Section 4.3).
4. Determine which of these cells have activity that is statistically significant (Section 4.4). Note that a cell might have some active compounds by chance and, because there are very many cells, multiplicity issue arises. We propose a permutation test to overcome this issue.
5. Score and prioritize untested compounds based on the good cells identified (Section 4.5). New compounds appearing frequently amongst the good cells are promising candidates for testing.

4.1. Forming Subspaces and Cells

We use the data-driven binning method described by Lam et al. (2001) to divide a space into cells. Then we shift these cells in the various dimensions to allow for forming active regions of different shapes.

Binning the Descriptor Space into 1-D, 2-D, and 3-D cells.

The advantage of dividing a space into cells is that a number of methods can be developed to identify good cells, i.e., those with a high proportion of active compounds. It is also inherently local, allowing for the isolation of small active regions. We now review some methods for dividing a high-dimensional space into many small, low-dimensional cells.

In a conventional cell-based method, the range for each of the descriptors is subdivided into *m* bins of equal size. With the 67 BCUT descriptors, we would have m^{67} cells. Even with *m*=2, there are 2^{67} (or $1.5\text{x}10^{20}$) cells generated, most of which are empty even for the largest ever-existing chemical database. There would be more cells than data points. In addition, most compounds will be densely clustered in relatively few cells, making it difficult or impossible to separate active and inactive regions.

Following Lam et al. (2001), we focus our attention on low-dimensional subspaces, typically all 1-D, 2-D, and 3-D subspaces. This strategy is motivated by Pearlman and Smith's (1999) "receptor-relevant subspace" concept. They argued that often only two or three BCUT descriptor variables are important for activity against a particular biological receptor and that activity is highly localized within the relevant subspace. Secondly, we keep the number of cells constant over each subspace, avoiding the exponential increase in the number of cells with dimension. Consequently, the (average) number of compounds per cell does not decrease with dimension, maintaining statistical power for separating active and inactive regions (cells). Furthermore, if only a few descriptors are relevant for a particular mechanism, some low-dimensional cells containing only important variables are likely to be identified, facilitating understanding. In contrast, higher-dimensional cells would include unimportant variables. To keep the number of cells constant, higher-dimensional cells would also have to be larger in the subspace of important variables, possibly too large to isolate a localized active region. Lastly, to avoid empty cells caused by the scarcity of molecules towards the limits of a descriptor's range, we adopt a data-driven hybrid binning method that makes bins larger towards the extremes.

Briefly, cells are created as follows. Initially, we divide each descriptor into *m* bins. For each descriptor, these bins are immediately the cells for its 1-D subspace. To form the cells for a given 2-D subspace, amalgamate the *m* 1-D bins into $m^{1/2}$ larger bins for each of its dimensions. There are $m^{1/2}$ x $m^{1/2} = m$ 2-D cells from combining these larger bins. Similarly, to form 3-D cells, we amalgamate each dimension's 1-D bins into $m^{1/3}$ bins; these are combined to give $m^{1/3}$ x $m^{1/3}$ x $m^{1/3} = m$ 3-D cells. Thus, all subspaces, whether 1-D, 2-D, or 3-D, have the same number of cells. To generate integer numbers of bins, it is convenient if *m* is an integer raised to the power of 6, e.g., 2^6 =64 or 4^6 =4096. We give further guidance below on choosing *m*. For more details in binning a high dimensional space into low-dimensional cells see the sections 'Forming Cells' and 'Data-Driven Binning' in Lam et al. (2001).

With *k* descriptors, there are

$$\binom{k}{1}+\binom{k}{2}+\binom{k}{3}=\frac{5}{6}k+\frac{1}{6}k^3$$

1-D, 2-D, and 3-D subspaces in total. For every subspace, a molecule is in one and only one cell. The goal is to find a set of cells in which there are many active compounds and a high proportion of active compounds.

How large should the bin size be? Cells formed from large bins may contain more than one class of compounds. Moreover, if only part of the cell is good, active compounds will be diluted by inactive compounds and the cell may be deemed inactive. (Two compounds must have fairly close values of all critical descriptors for similar biological activity.) On the other hand, a cell formed by very fine bins may not contain all the compounds in the same class. Furthermore, very small cells will tend to have very few compounds and there will be little information to assess the quality of the cell. We make the bins fine, but not too fine, given *N*, the number of assayed compounds. For reliable assessment of each cell's hit rate, we would like at least 10 compounds per cell. This suggests that the number of cells per subspace should be no more than *N*/10.

Intra-subspace cells (not including the shifted cells described below) within a subspace are mutually exclusive and cover different sets of compounds. On the other hand, inter-subspace cells, cells from different subspaces, can cover the same set of compounds. The compound-selection method described in Section 4.5 takes advantage of the collective strength of inter-subspace cells and makes use of the small amount of extra information available when further highly correlated descriptors are added.

Shifted Cells

The data-driven binning method generates non-overlapping cells within a subspace. We call these the original, unshifted cells. Because the location and the shape of an active region are unknown, it is not possible to define the exact boundaries of a cell that perfectly fit an entire active region. The cell boundaries are fixed prior to analysis. For example, an active 2-D region with four active compounds can be sliced, by chance, into four 2-D cells with one active compound in each cell. In this case, none of the four 2-D cells will be identified as good cells and thus the active region will not be found.

To allow for the fact that the original binning may not be optimal, we also shift the original cells in the various dimensions to create overlapping cells (shifted cells). For example, Figure 3 shows the locations of 10 active compounds in the subspace formed by two descriptors, x_1 and x_2. To form 2-D cells, the range of each descriptor is divided into five bins here. We generate four sets of cells: one set of original, unshifted cells, two sets of cells with only one dimension shifted by half a bin, and one set of cells with both dimensions shifted half a bin. These four sets of cells are shown in Figures 3(a)-(d), respectively. The good cells identified in analysis are then used to form active regions. If a good cell has to have at least three active compounds (as in Section 4.2), there is one active cell in each of Figures 3(a) and 3(b) and there are two active cells in each of Figures 3(c) and 3(d). The region formed by these overlapping active cells is shown in Figure 4. The counts are the number of times each active compound falls in an active cell. The dashed lines show how the active region could be adjusted to exclude sub-regions with no actives. Note that parts of the active region missed by the original binning are found.

The shifted cells provide an effective means of handling different shapes of active regions, at the price of looking at more cells. The number of cells created for a d-dimensional subspace is increased by a factor of 2^d and the number of bin cut-off points for each dimension is doubled. For example, if a 3-D subspace is divided into 4×4×4 = 64 cells, shifting will lead to a total of 8×64 cells, which is as many as an 8×8×8 arrangement. Therefore, this method allows us to use larger bins for the analysis, with more compounds per cell, and hence higher power for detecting activity.

We also investigated several methods for determining the shape and the size of an active region. However, we found that growing and shrinking a cell around an original, active cell to cover adjacent active cells was more complex and not as effective and efficient as shifting cells.

4.2. Preliminary Identification of Good Cells

We make a preliminary reduction of the huge number of cells that can be generated, particularly when there are many descriptors. We search every cell and note the ones with several (say three) active compounds. These are preliminary good cells. Then we adjust the boundaries of the preliminary good cells to exclude sub-regions with no active compounds. In later stages of the analysis, the hit rate and other related statistics will be computed for each of the re-sized cells. Those cells with a low proportion of active compounds will be removed. Active regions will be created by combining the remaining good cells.

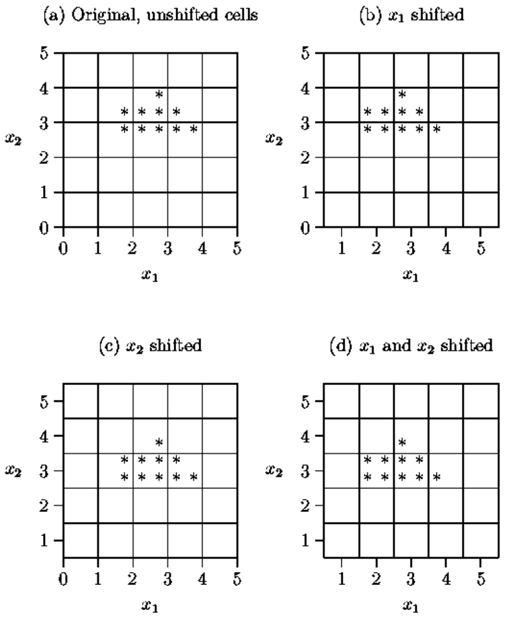


Figure 3. Shifted Bins (Five per Descriptor) and Overlapping Cells for 10 Active Compounds in a 2-D Subspace Formed by x_1 and x_2 : (a) original, unshifted cells; (b) only the x_1 bins are shifted by half a bin; (c) only the x_2 bins are shifted by half a bin; and (d) both the x_1 and the x_2 bins are shifted by half a bin.

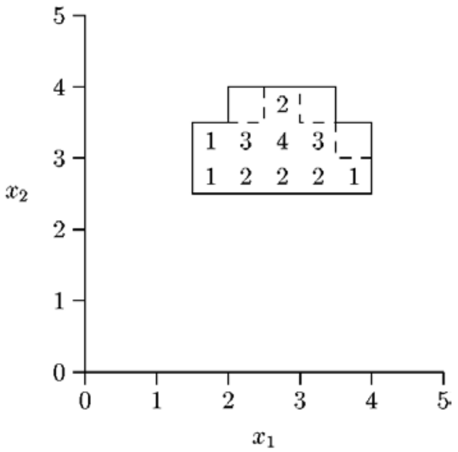


Figure 4. Overlapping Shifted Cells to Form an Active Region. The box denotes the active region. The counts are the number of times each active compound is selected by active cells (those with at least three active compounds). The dashed lines show how the active region could be adjusted to exclude sub-regions with no actives.

Preliminary Good Cells

After the (original and shifted) cells are constructed, the next step is to search for preliminary good cells: those with at least a certain number of active compounds. The required number of active compounds will depend on the total number of active compounds found in the data set. If only a few active compounds are available (e.g., less than 20), then all cells with two or more active compounds might be of interest. On the other hand, if there are hundreds of active compounds, then it is more efficient to pay attention to only those cells with, say, at least five active compounds. Of course one can examine every single cell with one active compound but this will generate many preliminary good cells by chance. For the examples described in Section 5, there are about 80 active compounds in the NCI training data set and about 40 active compounds in the Core98 training data set. In these examples, requiring two active compounds gives similar results to requiring three, but the latter generates fewer preliminary good cells.

The search for the preliminary good cells is straightforward. In principle, we just need to count the number of active compounds in every cell in every subspace. Because active compounds are usually rare in the data set, the search can be made computationally efficient by tracking them to the relatively few cells that they occupy. Subsequent stages of analysis are made much faster by working with the much-reduced list of preliminary good cells.

Re-sizing Cells

As the cell boundaries are fixed prior to analysis, a cell may cover both active and inactive regions and hence the observed hit rate of a cell can be misleading (active compounds may be diluted by inactive compounds, yielding a very low hit rate). To get a more focused region, we re-size each cell by trimming off the borders with no active compounds. Then, in each trimmed cell, we use the compounds remaining to determine the hit rate and other related statistics. These trimmed cells will be used later on to form active regions and to score and prioritize untested compounds for screening.

4.3 Ranking Cells

The next stage is to rank the re-sized cells (original and shifted). These rankings will be used in the later stage to score new compounds. All the ranking criteria are based on measures for individual cells.

With active/inactive binary-response data, a natural first choice for the identification of active cells is to compute the proportion of all the compounds in the cell that are active (the observed hit rate) and then rank the cells by these proportions. The main problem with this method is that it favors cells that happen to have a small number of compounds. Consider two cells with 2/2 and 19/20 active compounds, respectively. The first has a hit rate of 100%, but this is based on two compounds, a very small sample. The 95% hit rate for the second cell is based on 20 compounds and is much more reliable. Thus, in addition to the raw hit rate (*H*), we describe below two further criteria that take into account the statistical variability from sampling: p-value (*P*) and the binomial hit rate lower confidence limit (H_{L95}).

With a numerical assay value *Y* (e.g., percentage inhibition) for activity, we will similarly describe the raw mean activity score ($\overline{Y}$) and two criteria penalizing a small sample size: the lower confidence interval for the mean *Y* ($\overline{Y}_{L95}$) and the hit rate lower confidence limit based on a normal distribution for *Y* (H^{Y}_{L95}). Quantitative data of this type may also be converted to active/inactive classes by defining "Active" as $Y > c$ for some cut-off *c*, allowing all criteria to be used.

P-value (*P*)

Let *N* be the total number of compounds in a data set (e.g., 4096 compounds in the Core98 training set), and let N_a be the number of active compounds in the data set (e.g., 41 active compounds). Consider a given cell in a given subspace, which has *n* compounds, of which *a* are active.

Suppose the N_a active compounds are distributed such that they fall in or outside the given cell at random. Under this statistical null hypothesis, the probability of observing *a* actives out of *n* compounds is given by the hypergeometric distribution:

$$\text{Prob}\,(a;n,N_a,N)=\frac{\binom{N_a}{a}\binom{N-N_a}{n-a}}{\binom{N}{n}}$$

The p-value is the probability of having at least *a* active compounds out of *n*:

p-value = Prob($A \geq a$ | *n* compounds)

$$=\sum_{i=a}^{\min(N_a,n)}\frac{\binom{N_a}{i}\binom{N-N_a}{n-i}}{\binom{N}{n}} \quad = \quad 1-\sum_{i=0}^{a-1}\frac{\binom{N_a}{i}\binom{N-N_a}{n-i}}{\binom{N}{n}}.$$

If the p-value is small, there is little chance of seeing *a* or more active compounds out of *n*. Therefore, small P-values provide the most evidence against the null hypothesis of random allocation of actives in/outside the cell (and hence most evidence that the number of actives in the cell is better than chance). The P-value is computed for all cells and the cell with the smallest P-value is the top-ranked cell, etc.

The p-value approach tends to pick cells with large numbers of compounds even if they have fairly low hit rates. Suppose there are 40 active compounds in a data set of 4,000 compounds. Then 8 actives out of 80 (hit rate=0.10) gives $p=8.24\text{x}10^{-7}$ but 3 out of 3 (hit rate=1.00) gives $p=9.3\text{x}10^{-7}$. The statistical evidence is stronger in the first case because of the larger sample size, even though the hit rate is much lower. This illustrates the major drawback of the *P* criterion: it tests whether the hit rate is significantly larger than random, not whether the hit rate is large.

Hit Rate (*H*)

In the above notation, the hit rate for a cell is *a/n*. It ignores the increased reliability from a larger sample size. For example, 1/1 gives a 100% hit rate but 9/10 gives a 90% hit rate, yet the cell with 9/10 seems more promising. Although commonly used, it is not a sensitive criterion for ranking active cells (regions). The next criterion introduced considers both the hit rate and its variability.

Binomial Hit Rate Lower Confidence Limit (H_{L95})

One can obtain an exact lower confidence limit on the hit rate for new compounds based on the binomial distribution. For the many possible compounds that would fall in a given cell, suppose that a proportion *h* are active, i.e., *h* is the hit rate. Assuming that the *n* compounds in the cell that have been assayed are a random sample of all the cell's possible compounds, the number of actives, *A*, is a random variable following a binomial distribution with *n* trials and probability *h*. The smallest value of *h* such that $\text{Prob}(A \geq a \mid h, n) = 0.05$ is the 95% binomial hit rate lower confidence limit (H_{L95}). It considers both the hit rate and its variability. Some examples of cell rankings using the H_{L95} method are given in Table 2.

Table 2. Illustrative Cell Rankings Using H_{L95}.

Cell	*a* / *n* (Hit Rate)	H_{L95}	Ranking
1	9/10 (0.9)	0.6058	1
2	3/3 (1.0)	0.3684	2
3	8/80 (0.1)	0.0507	3
4	1/1 (1.0)	0.0500	4

Mean Activity Score ($\overline{Y}$)

When a numerical assay value, *Y*, is available, the mean over all compounds in a cell gives the mean activity score ($\overline{Y}$). Because it is easier by chance to obtain a high mean from fewer compounds than from more compounds, $\overline{Y}$ tends to pick cells with few compounds and high activity values. Although commonly used, it is not a sensitive criterion for ranking active cells (regions). The next criterion introduced considers both the observed mean and its variability.

Lower Confidence Limit for Mean Y ($\overline{Y}_{L95}$)

Analogous to H_{L95}, with a numerical assay value, *Y*, one can use the lower confidence limit for the mean of the distribution giving the *Y* values, based on an assumption of sampling from a normal distribution. This criterion, $\overline{Y}_{L95}$, considers both the observed mean and the variability and is defined as

$$\overline{Y}_{L95} = \overline{Y} - \hat{\sigma} / \sqrt{n} \times t(n-1, 0.95),$$

where, based on *n*-1 degrees of freedom, $\hat{\sigma}$ is the sample standard deviation within the cell and *t(n*-1*, 0.95)* denotes the 95% quantile of the *t* distribution.

Normal Hit Rate Lower Confidence Limit (H^{Y}_{L95})

With a numerical measure of activity, *Y*, and a cut-off for activity, *c*, one can derive a lower confidence limit for the hit rate, i.e., the probability Prob(*Y*>*c*), based on the assumption that the observed activities in a cell are randomly sampled from a normal distribution. This criterion is called H^{Y}_{L95}.

If the *Y* values are randomly sampled from a normal distribution with mean μ and variance σ^2, then by definition, H^{Y}_{L95} is

$$\Pr(Y > c) = 1 - \Pr(Y \le c) = 1 - \Pr\left(\frac{Y-\mu}{\sigma} \le \frac{c-\mu}{\sigma}\right) = 1 - \Phi\left(\frac{c-\mu}{\sigma}\right) = \Phi\left(\frac{\mu - c}{\sigma}\right),$$

where Φ is the standard normal cumulative distribution function.

Suppose σ is known or a good estimate is available (the pooled estimate described below will usually have many degrees of freedom). Then we can estimate Φ by

$\hat{\Phi} = \Phi\left(\frac{\overline{Y} - c}{\sigma}\right)$, where $\overline{Y}$ is the average *Y* value for the *n* compounds in the cell.

Let $Z = \frac{\mu - c}{\sigma}$, which we estimate by $\hat{Z} = \frac{\overline{Y} - c}{\sigma}$. We have $E(\hat{Z}) = \frac{\mu - c}{\sigma}$ and $Var(\hat{Z}) = \frac{\sigma^2}{n}\frac{1}{\sigma^2} = \frac{1}{n}$.

Therefore,

$$\hat{Z} \sim N\left(\frac{\mu - c}{\sigma}, \frac{1}{n}\right) \quad \text{and} \quad \Pr\left(\frac{\hat{Z} - \frac{\mu - c}{\sigma}}{1/\sqrt{n}} < Z_{.95}\right) = 0.95,$$

where $Z_{.95}$ is the 95% quantile of the standard normal distribution.

Rearrangement of the inequality gives

$\Pr\left(Z_L < \frac{\mu - c}{\sigma}\right) = 0.95$, where $Z_L = \hat{Z} - \frac{Z_{.95}}{\sqrt{n}}$.

A 95% lower confidence interval (CI) for $\frac{\mu - c}{\sigma}$ is (Z_L, ∞) and the corresponding 95% CI for $\Phi\left(\frac{\mu - c}{\sigma}\right)$ is $(\Phi(Z_L), 1)$ since Φ is a monotonic increasing function.

Therefore, H_{L95}^{Y} can be estimated by $\Phi(Z_L) = \Phi\left(\frac{\overline{Y} - c}{\sigma} - \frac{Z_{.95}}{\sqrt{n}}\right)$.

We use a common estimate of σ for all cells within a subspace. For a given subspace, it is computed by pooling the sample variances over all cells:

$$\hat{\sigma}^2 = \frac{\sum (n_i - 1) s_i^2}{\sum (n_i - 1)},$$

where s_i^2 is the sample variance for cell *i,* and cell *i* has n_i compounds.

Relationships between the criteria

If a numerical measure of activity is available, all six criteria can be used. The cut-point *c* for activity (a hit) is used as follows. For *P*, *H* and H_{L95}, *c* is used to convert the data to “Active” / “Inactive” before they are computed. Both $\overline{Y}$ and $\overline{Y}_{L95}$ ignore *c*. For H_{L95}^{Y}, the *Y* distribution is modeled and *c* is used at the end to determine H_{L95}^{Y}.

4.4 Assessing the Impact of Multiple Testing

With 67 descriptors, there are a total of 50,183 1-D, 2-D, and 3-D subspaces. If each subspace is divided into 64 cells and the cells are shifted in the various dimensions (see Section 4.1), there are 25,101,952 (shifted and unshifted) cells. With so many cells, it is possible that by chance alone we will see cells with moderate activity.

Consider the p-value criterion. To adjust it for the total number of cells examined, C, we simply multiply each p-value by C. This is the Bonferroni correction (Miller 1981). In the training data, a cell is said to be a good cell by the p-value criterion if the Bonferroni adjusted P is small (say <0.05).

The Bonferroni correction tends to over-correct, but we can impose a minimum number of active compounds to define the cells relevant for correction. In the NCI example with 67 BCUTs and 25,101,952 cells, for example, only 5,587,591 cells have at least two active compounds, a smaller adjustment factor.

Probably the best way of addressing the multiple testing problem is to define the cut-off between active and inactive cells using a random permutation of the assay values. The Active/Inactive indicators or Y values in the training data are randomly reordered, i.e., randomly assigned to the descriptor combinations in the data set. If p-value is the criterion for ranking cells, one can set the cut-off as a small p-value in the lower tail of the distribution induced by randomization. Under random permutation of the data, no cells should be identified as good cells and the smallest p-value is just due to chance. For the actual data (without permutation) one can then use all cells with p-value smaller than this cut-off point.

Ideally, to estimate the p-value corresponding to a true significance level of say 5%, we would like to perform many random permutations. The sets of p-values from these randomizations would be combined into an empirical distribution, and the 5% point from this distribution is a multiplicity-adjusted critical value. This is too computationally expensive, however. Fortunately, for the cell-based analysis, one permutation provides many p-values (e.g., 25,101,952 cells and hence p-values). Thus, we take the 5% point from one permutation as the cut-off to determine whether there are any real active cells (versus false alarms). This procedure can be applied to any of the cell-ranking criteria in Section 4.3.

Cells with ranking scores in the actual data that beat the random-permutation cut-off are used to score and select new compounds. The subspaces and descriptor ranges associated with these cells indicate descriptors that are likely relevant to activity and subregions of activity, respectively. New compounds appearing in the most highly ranked cells or frequently amongst the good cells are promising candidates for testing, as described next.

4.5. Selection of New Compounds

We present three selection methods for choosing untested compounds for biological screening: 'Top Cells Selection', 'Frequency Selection' and 'Weighted Score Selection.' All the methods first rank cells according to one of the criteria in Section 4.3 and apply the random-permutation method of Section 4.4 to generate a list of good cells.

Top Cells Selection

In a database of new, unassayed compounds, top-cells selection chooses all the compounds falling in the best cell, then all those in the second best cell, and so on until the desired number of compounds to be tested is reached or until there are no good cells remaining. This approach does not combine strength from several good cells when scoring a compound. The next method takes advantages of the collective strength of the good cells, thus increasing the prediction power.

Frequency Selection

Frequency selection scores a new compound by the number of times it appears in the list of highly ranked cells. The first compound selected for screening is the one occurring with the maximum frequency, the second compound selected has the second largest frequency, and so on.

Frequency selection scores a compound based on many good cells and possibly many descriptors. A single cell belongs to a subspace involving only one, two or three variables, and cells are scored individually. In contrast, under frequency selection, if a new compound appears in several good cells in different subspaces, information is combined from the union of all the subspaces' descriptors. Thus, frequency selection can potentially make use of the small amount of extra information available when further highly correlated descriptors are added (see the comparison of 6 and 67 descriptors in Section 5.3).

Frequency selection provides a powerful way to rank new compounds for screening, often leading to a very high hit rate for the top ranked compounds. The next method introduced further improves the compound ranking by incorporating information on the order of the cells in the list.

Weighted Score Selection

Instead of just counting the frequency of occurrence in the list of good cells, we can give each cell in the list a weight and score a new compound based on the total weight over the cells in which it resides.

The cell-ranking criteria described earlier can be adapted as weight functions. We could use the H_{L95} value or –log(p-value) as weights, for example. The weight function should have several desirable properties: (1) If the list of good cells is extended, the relative weights of the cells in the original list should not change; (2) the weight function should be a smooth monotonic decreasing function of the cell's rank; and (3) the same weight should be assigned to cells rated equally by the cell ranking criterion. For the numerical evaluations in Section 5, we use weighted score selection with $\boldsymbol{H_{L95}}$ (NCI binary-response data) or $\overline{Y}_{L95}$ (Core98 continuous-response data) values as weights. These are the criteria used for cell ranking to generate the list of good cells.

5. Performance Evaluation

We evaluate the performance of our cell-based analysis method using the 23056 Core98 compounds and the 29812 NCI compounds. The objective of this evaluation is (1) to determine if the new methods lead to higher hit rates than random selection, (2) to assess the effect of the six cell selection criteria on hit rate, and (3) to determine whether our cell selection method can find real active cells or false alarms.

In addition, we compare the cell-based analysis method with recursive partitioning (the tree function in S-Plus, Clark and Pregibon 1992) in terms of identifying active compounds. Often, V-fold cross-validation is used to control tree size, but here this tends to result in a very small tree, sometimes with only a root node. This problem seems to arise because active compounds are rare in the training data, and the smaller hold-out samples have too few active compounds to compare different tree sizes. For simplicity, then, we use the default S-Plus tree (from default fitting options, e.g., minimum 5 observations per node) and do not attempt to prune this tree. (Some preliminary work by graduate student Marcia Wang also suggests that tree pruning is ineffective anyway.)

5.1. Evaluation Plan

To evaluate the cell-based analysis method for the two data sets, we carry out the following steps.

1. Divide the data into Training and Validation sets. Samples of 4096 compounds are selected to form the training data set; the rest of the compounds form the validation data set. Samples are chosen randomly or using uniform coverage designs (Lam et al. 2001).

2. Apply the data-driven hybrid binning method to bin all subspaces, and create 64 cells per subspace. This gives 64 compounds per cell on average. Create shifted cells from the original bins (Section 4.1).
3. Training set: Search for preliminary good cells with two or more active compounds (Section 4.2).
4. Training set: Compute summary statistics for the preliminary good cells: H_{L95} for the NCI binary-response data or $\overline{Y}_{L95}$ for the Core98 continuous-response data (Section 4.3). Perform a permutation test (Section 4.4) to find the cut-off point to separate good cells from false alarms. This generates a list of good cells (considered as 'real').
5. Validation set: Score and select new compounds from the validation set based on the good cells identified from the training set. Here we can rank the validation-set compounds using weighted score selection (Section 4.5).
6. Validation set: As compounds are successively selected, evaluate the hit rate (binary response) or mean activity value (continuous response) as performance measures.

We first look in detail at the multiplicity correction in Step 4, then present the final hit rate and mean activity performance results.

5.2. Good Cells Versus False Alarms

Bonferroni Correction

To test whether our cell-based method would give false-positive results, the activity values are randomly re-assigned to the compounds. The cell-based analysis is carried out on the permuted data. Using the p-value correction method described in Section 4.4, few cells are declared good. On the other hand, many good cells are found using the real activity values. This approach addresses the false positive problem, but is probably quite conservative.

Permutation Test

To illustrate how the permutation test in Step 4 works, we examine a random sample of 4096 compounds from the NCI data set with 67 descriptors. For this sample, we generate 25,101,952 cells (see Section 4.4) and analyze these cells twice, once with the original activity values and once with randomly re-arranged activity values. Under random permutation, the best cell had 7 out of 7 active compounds, with P=2.29×10^{-12} and H_{L95}=0.6518, as shown in Table 3. With the real data, Table 3 also shows there are 5,256 cells with a smaller p-value and 449 cells with a larger H_{L95} value, suggesting that these cells are indeed good active regions and that the descriptors are relevant to the activity.

Table 3. P and H_{L95} Values for Different Cut-Off Points and the Corresponding Number of Cells with a Better Value.

	Under Random Permutation Criterion value (#actives/#compounds)		Real Data #cells with better value	
	P	H_{L95}	P	H_{L95}
Best value	2.29E-12 (7/7)	0.6518 (7/7)	5,256	449
The 5% point	1.04E-4 (6/35)	0.2236 (2/2)	782,864	493,962

For defining the list of good cells and hence selecting new compounds for screening, we use a less conservative cut-off: the 5% point of the distribution under randomization. Using the 5% point, many more cells are found with better values. Collectively, these cells enhance the prediction power of the compound selection methods. Scoring new cells using weighted frequency of occurrence in the list of good cells (Section 4.5) is insensitive to adding some possibly spurious cells to the bottom of the list: these cells have low weights.

In this example, two practical issues are also revealed. As mentioned earlier, the raw hit rate H is not a sensitive cell-ranking criterion, and the permutation test based on H often leads to a hit rate cut-off at 100%, making identification of good cells difficult. Also, one has to be careful in the implementation of the P criterion as the p-values for good cells can be extremely small. In addition, P tends to pick cells with large numbers of compounds (Section 4.3), hence our use of H_{L95} as the primary ranking criterion for data with binary response. Similar comments apply to the Core98 continuous-response data and our preference for the $\overline{Y}_{L95}$ criterion.

5.3. Validation Hit Rates

A total of 80 training sets were generated, 40 from each of the NCI and Core98 data sets. Half of the training sets were generated using random selection and the other half were generated using uniform coverage designs (Lam et al., 2001). For comparisons between the cell-based (CB) analysis method and recursive partitioning (RP), the S-Plus classification and regression tree method with default settings is used (Venables and Ripley, 1999). Except where we specifically compare 6 and 67 descriptors, all analyses are performed using the original 6 descriptors to reduce the burden of computational effort.

Cell-Based Analysis Versus Recursive Partitioning

Twenty training sets of 4096 compounds were randomly generated from each of the NCI and Core98 compounds. These training sets were analyzed using both CB and RP analysis methods. The mean hit rates and mean activity results based on the validation sets are shown in Figure 5. For the NCI compounds, the CB analysis clearly dominates the RP analysis. Both methods generate hit rates many times higher than the random hit rate. For the Core98 compounds, the CB analysis outperforms the RP analysis for the first 50 compounds selected; thereafter the two methods are comparable. Again, both methods perform much better than the random-activity baseline. The Core98 activity values have much larger measurement error than the NCI activity; in addition, the Core98 compounds have fewer hits. Both of these facts make predicting active compounds difficult.

Impact of Design on Cell-Based Analysis: Uniform Coverage Designs Versus Random Selection

Here we investigate the impact of different designs for the training data on the performance of the CB analysis. Two types of designs are compared: simple random sampling (as in the CB versus RP comparison) and uniform coverage designs (Lam et al., 2001). By keeping the sample size within each cell fairly constant, the uniform coverage designs should provide good power across all cells. Twenty training sets of 4096 compounds are generated, using the two methods, from each of the NCI and Core98 data sets. These training sets are analyzed using the CB analysis method. The mean hit rate and activity results are shown in Figure 6. Using the uniform coverage designs, additional improvement in hit rate or mean activity is found for the first 100 compounds selected. The bumps in Figure 6(a) when only 1-25 compounds are selected are likely due to the discreteness of the binary response: a few extra hits will make a big impact on the hit rate.

Six Versus 67 Descriptors

Because of high computational cost, only two samples from the 20 random training sets for the NCI compounds are chosen to evaluate the information gain from using more BCUT descriptors. The two samples have the highest and lowest validation hit rates at the 100th compound selected in the six-descriptor cell-based analysis: 74/100 hits and 47/100 hits, respectively. Re-analysis of the same two samples using the 67 descriptors gives the hit rate results shown in Figure 7. In both samples, the 67 descriptors lead to higher hit rates for the CB analysis. The CB analysis gains predictive power despite the strong correlations among the descriptors. This is not so for the RP analysis. The hit rate results at 100 compounds selected are summarized in Table 4.

Figure 7 also indicates that CB analysis is fairly robust to variability due to random sampling. Designs generated by different random samples will lead to training data with little overlap. Therefore, CB analysis will probably be working with rather different sets of preliminary good cells, cell scores, and compound scores. Nonetheless, as Figure 7 shows, the hit-rate performance is similar, especially if 67

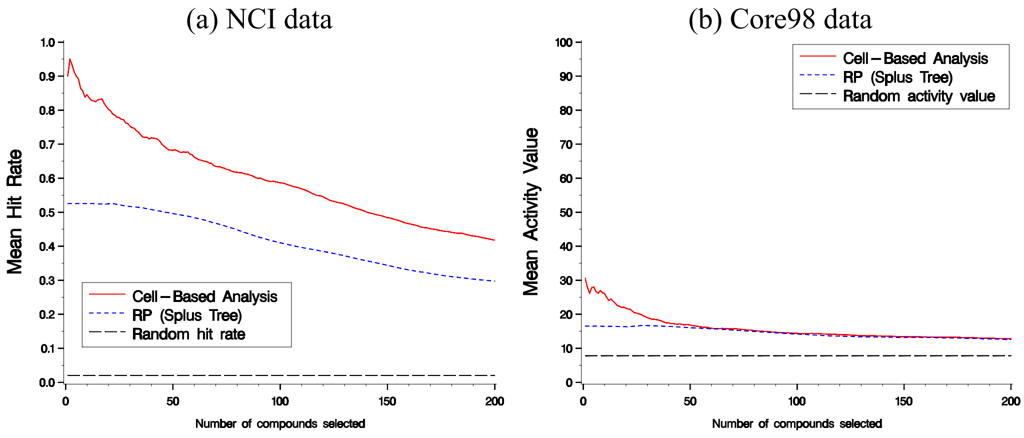


Figure 5. Average Performance of Cell-Based Analysis (Solid Line) and Recursive Partitioning (Dashed Line) for 20 Random Samples When the 200 Validation-Set Compounds With the Highest Scores Are Selected: (a) Mean Hit Rate for the NCI Binary Data and (b) Mean Activity for the Core98 Continuous-Response Data. The horizontal line near the bottom shows the expected performance under random selection of new compounds.

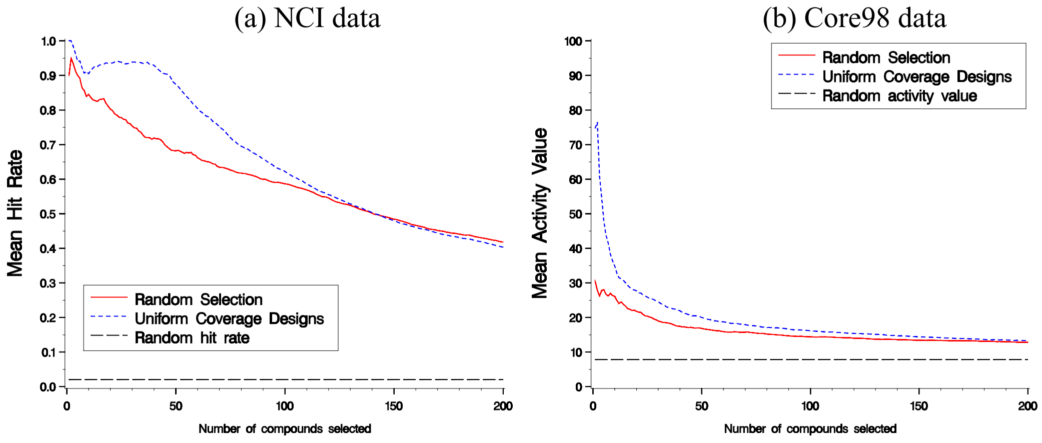
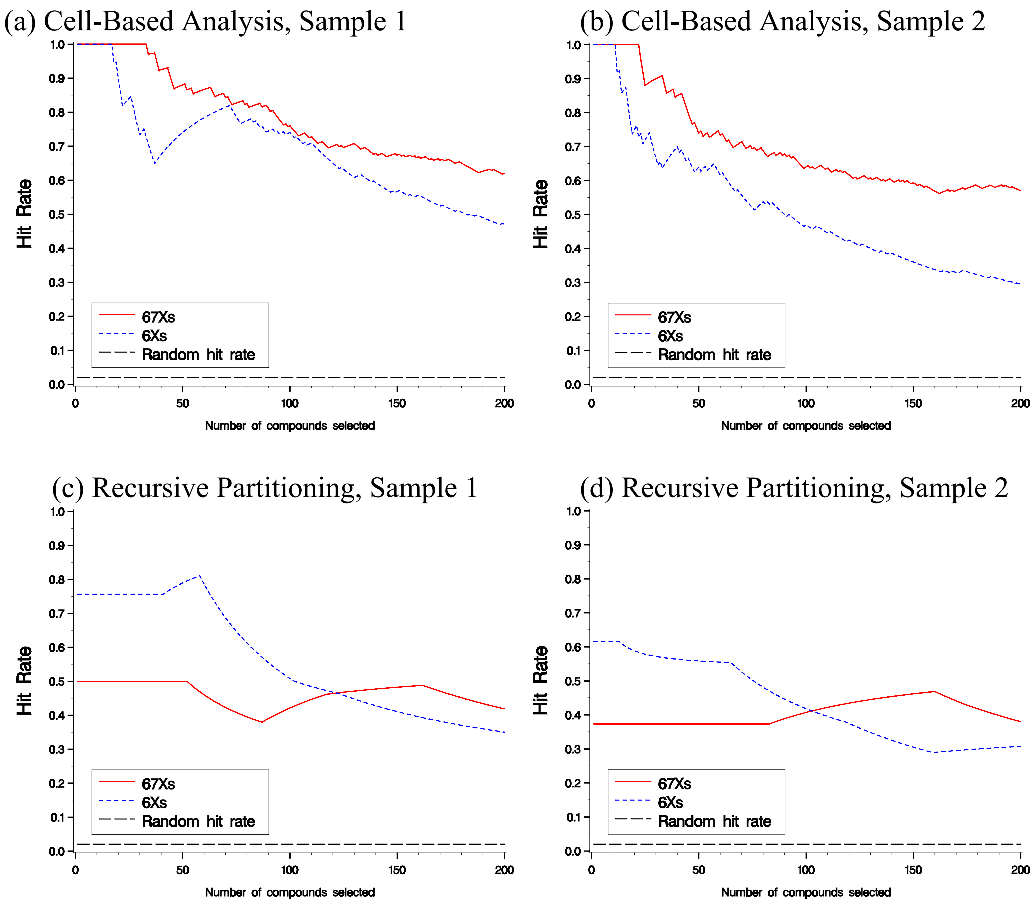


Figure 6. Average Performance of 20 Random Designs (Solid Line) and 20 Uniform Coverage Designs (Dashed Line) When the 200 Validation-Set Compounds With the Highest Scores are Selected By Cell-Based Analysis: (a) Mean Hit Rate for the NCI Binary Data and (b) Mean Activity for the Core98 Continuous-Response Data. The horizontal line near the bottom shows the expected performance under random selection of new compounds.



(a) Cell-Based Analysis, Sample 1

(b) Cell-Based Analysis, Sample 2

(c) Recursive Partitioning, Sample 1

(d) Recursive Partitioning, Sample 2

Figure 7. Hit Rates for Two Random Samples from the NCI Data Using Either 6 Descriptors (Solid Line) or 67 Descriptors (Dashed Line). The figure also compares CB and RP analyses.

descriptors are used. The differences between the hit-rate profiles for the two samples are small here relative to the differences between CB analysis and recursive partitioning. In general, CB analysis is not likely to be sensitive to small changes in the data (e.g., adding or removing a few compounds), because such changes will only affect a few cells and the method is inherently local.

Table 4. Hit rates, at the 100th Compound Selected, by Different Analysis Methods and by Different Sets of Descriptors.

	Cell-Based Analysis		Recursive Partitioning	
Sample	6 BCUTs	67 BCUTs	6 BCUTs	67 BCUTs
1	0.740	0.760	0.508	0.421
2	0.470	0.640	0.418	0.407

6. Conclusions and Discussion

These results confirm that (1) the cell-based analysis method is useful in identifying good cells, (2) many good cells are found, not false alarms, and (3) the BCUT descriptors are informative. Our cell-based analysis method leads to hit rates many times higher than the random hit rate. It consistently leads to very high hit rates for the top ranked compounds. To get a sense of the possible increases in efficiency, consider the following. Using random screening, one would expect to screen 1,000 NCI compounds to find 20 active compounds; however, using the CB prediction one can identify 20 active compounds by screening just 20 compounds: see the curves for 67 descriptors in Figures 7(a) and 7(b). The CB prediction method compares favorably with RP here.

In principle, because it is inherently local, a cell-based analysis should be able to handle nonlinear, threshold, and interaction effects as well as multiple activity mechanisms. By combining scores from many cells (low-dimensional projections) it should also be able to extract further information from highly correlated descriptors.

On the other hand, linear regression models are not effective in handling these modeling issues. For illustration, polynomial regression models of degree 3 including interaction terms of 2 and 3 descriptors were fitted to the Core98 data using the stepwise-selection method. The ‘best’ model had $R^2 = 0.01$ and poor prediction accuracy in identifying compounds as active. For the NCI data, logistic regression models were also investigated. Overall, low prediction accuracy in classifying compounds as active and high prediction accuracy in classifying compounds as inactive were found. As only about 2% of compounds are active, any methods claiming all compounds as inactive will give an overall accuracy of 98%. The real challenge is to find a high proportion of active compounds.

Our goal is to find a set of regions (cells) in which there is a high proportion of active compounds. It is much easier to divide and cover low-dimensional subspaces and to identify low-dimensional active cells. Whereas RP evaluates one descriptor at a time, CB analysis evaluates 1-D, 2-D and 3-D cells (i.e., evaluates one, two and three descriptors at a time) and combines these cells when scoring to form high-dimensional active regions. Furthermore, the low-dimensional cells are formed from all combinations of different subsets of descriptors, so all descriptors can be effectively evaluated and the impact of irrelevant variables on analysis is reduced or eliminated. Therefore, focusing on low-dimensional subspaces is effective in finding active and inactive regions (cells).

Shifted cells provide an efficient and effective method for handling different shapes of active regions. In combination with re-sizing of cells, the boundaries of active regions can be better aligned. In compound selection, a compound appearing in more than one cell within the same subspace will be counted only once to avoid over-counting from the shifted cells. This is analogous to (1) forming an active region within a subspace, and (2) ranking the new compounds based on their (weighted) frequency in all active regions across all subspaces.

Designed experiments (e.g., uniform coverage designs) can enhance the predictive power of cell-based analysis. The actual improvement in prediction can be much greater and can be better evaluated if a real test set (instead of a hold-out set for validation) is available, as compounds in the hold-out set are not always available in every cell identified from the training set. Uniform coverage designs tend to select roughly the same number of compounds from both crowded and sparse regions and might not leave compounds in the sparse regions for validation.

A good prediction method should obtain more hits for the highest ranked compounds. Because the total number of hits is a constant, the hit rate or the activity value typically decreases as the number of tested compounds increases, all the way down to the random rate when all compounds are tested. The CB analysis method is particularly effective in finding hits when few compounds are selected.

We primarily used H_{L95} for binary response data and $\overline{Y}_{L95}$ for continuous response data. These criteria take account of uncertainty from the sample size and have fewer assumptions.

CB analysis is a multi-stage automated analysis process, which requires extensive computing power. There are many opportunities to make the algorithm more efficient as well as to further enhance the prediction accuracy. We are currently investigating these opportunities. One can use a combination of different ranking criteria (e.g., the P and H_{L95} values) to define a 'common' cut-off or even to select multiple sets of good cells (different criteria may select different types of active cells). For a very large data set (e.g., millions of compounds with many descriptors), a fast algorithm to store and evaluate billions of cells is needed. Tuning parameters such as the minimum number of active compounds required for a preliminary good cell, choosing cut-offs for the good cells, and more sensitive weighting functions for scoring cells and hence compounds, will be studied. The current cell re-sizing method (Section 4.2) re-sizes each cell by trimming off the borders with no active compounds. This is done by setting the new boundaries of a cell to the descriptor ranges of the active compounds. The more active compounds available in the cell, the better the boundaries can be located. Using this simple re-sizing method alone, without the shifted cells method, may leave holes within an active region. The shifted and unshifted cells overlap each other, thus reducing or minimizing possible holes in an active region. Other cell re-sizing methods will be investigated.

ACKNOWLEDGMENTS

We are grateful for the help of Eugene Stewart in providing the chemical descriptors. Welch's research was funded by GlaxoSmithKline and NSERC of Canada.

References

Bayley, M.J. and Willett, P. (1999) Binning schemes for partition-based compound selection. *Journal of Molecular Graphics and Modeling* 17, 10-18.

Breiman, L., Friedman, L., Stone, C.J. and Olshen, R.A. (1984) Classification and Regression Trees. Chapman and Hall.

Burden, F.R. (1989) Molecular Identification Number for Substructure Searches. *Journal of Chemical Information and Computer Sciences* 29, 225-227.

Clark, L.A. and Pregibon, D. (1992). Tree-Based Models, in Statistical Models in S. J.M. Chambers, and T.J.Hastie, eds. CRC Press, Boca Raton, Florida.

Hawkins, D.M. and Kass, G.V. (1982). Automatic Interaction Detection. In *Topics in Applied Multivariate Analysis*; Hawkins, D.M., Ed., Cambridge University Press, UK. pp 269-302.

Hawkins, D.M., Young, S.S., and Rusinko, A. (1997) Analysis of a large structure-activity data set using recursive partitioning. *Quant. Structure-Activity Relationship*, 16, 296-302.

Higgs, R.E., Bemis, K.G., Watson, I.A. and Wike, J.H. (1997) Experimental Designs for Selecting Molecules from Large Chemical Databases. *Journal of Chemical Information and Computer Sciences* 37, 861-870.

Jones-Hertzog, D.K., Mukhopadhyay, P., Keefer, C., and Young, S.S. (2000) Use of Recursive Patitioning in the Sequential Screening of G-protein Coupled Receptors. *Journal of Pharmacological and Toxicological Methods*, 42, 207-215.

Lam, R.L.H., Welch, W.J., and Young, S.S. (2001) Uniform Coverage Designs for Molecule Selection. paper submitted to *Technometrics*.

McFarland, J. W. and Gans, D.J. (1986) On the Significance of Clusters in the Graphical Display of Structure-Activity Data. *Journal of Medicinal Chemistry*, 29, 505-514.

Miller, R.G. (1981) Simultaneous Statistical Inference. 2nd Ed. Springer-Verlag, New York

Pearlman, R.S. and Smith, K.M. (1998) Novel software tools for chemical diversity. *Perspect. Drug Discovery Design* 09/10/11 339-353.

Pearlman, R.S. and Smith, K.M. (1999) Metric Validation and the Receptor-Relevant Subspace Concept. *J. Chem. Inf. Comput. Sci.* 39, 28–35.

Venables, W.N. and Ripley, B.D. (1999) Modern Applied Statistics with S-PLUS. 3rd Ed., New York, Springer.

Rusinko, A, III, Farmen, M.W., Lambert, C.G., Brown, P.L., and Young, S.S. (1999) Analysis of a large structure/biological activity data set using recursive partitioning. *Journal of Chemical Information and Computer Sciences*, 38, 1017-1026.

Young, S.S. and Hawkins, D.M. (1998) Using Recursive Partitioning to Analyze a Large SAR Data Set. *Structure-Activity Relationship and Quant. Structure-Activity Relationship*, 8, 183-193.