Supporting Information

Part I.
Sample Implementation of the Compound Acquisition and Prioritization Algorithm

The reported compound acquisition and prioritization algorithm was originally implemented in C++. For the sake of data portability, the raw data was imported and saved as "R" workspace ("R" is free software, available at http://cran.r-project.org/), and the scripts for compound acquisition and prioritization algorithm were written correspondingly. The "R" workspace and sample scripts were then compressed into the zip file (available at supporting information). The workspace and scripts were generated by "R" 2.8.0 for Windows platform. So any newer version of Windows-based "R" is recommended, although the work may be repeated by other versions of "R" as well.

File "RawData.RData" is a workspace file that has three variables: APL, mlscn, DistTc. Command "ls()" can be entered to list variables in the current workspace. Note: the variable names are case-sensitive. APL is a 1993 × 5 data frame. The first column lists the compound names for the Active Probes Library. The next four columns contain the BCUT partial charge, HBA, HBD and polarity descriptors respectively (Table 1). Similarly, variable mlscn saves compound names and BCUT descriptors for the existing compound collection. Variable DistTc is used to conduct regression analysis (Figure 4). Each row of DistTc includes the names of an APL compound pair (column 3 and 4), the Tanimoto coefficient between the compound pair (column 1), and the Distance between the two compounds in the BCUT chemistry space (column 2).

Three "R" scripts in the zip archive can be used to repeat the computation. First, run the script named "script_regression_analysis.txt" after opening workspace "RawData.RData". This script shows the result of the weighted regression analysis and plots Figure 4 automatically. Coefficients of the regression equation and R-square value can be found from the output. Then, run the script named "script_mlscn_nearest_neighbor.txt". This step aims at calculating the distance to the nearest neighbors for each compound in the existing compound collection. The calculation is slow, as it has $O(N^2)$ time complexity (N = 230000). The original C++ program implemented kd-tree to speed up nearest neighbor search. However, the results have been saved in workspace "MLSCN Nearest Neighbor Search.RData". Users can skip this step and retrieve the results by restoring the workspace. Finally, run the script with the name of "script_acquisition.txt". This script calculates the density of the existing compound collection and plots the histogram of distances between nearest neighbors (Figure 5). The density of the existing compound collection is saved in variable "lambda". The script further outputs the names of to-be-acquired compounds according to the compound acquisition and prioritization algorithm.

The compound acquisition and prioritization algorithm is not difficult to implement in practice. The scripts provided here exhibit sample usage of the algorithm. These scripts partially repeat the work reported in the manuscript, even if only Active Probes Library is demonstrated.

Part II.

In the compound acquisition and prioritization algorithm, compound selection is carried out using BCUT descriptors. However, it may be necessary to remove the compounds with undesired properties from the to-be-acquired subsets explicitly by applying some other criterions. The choice of the criterions depends on what these compounds are used for in the future studies. We illustrate this procedure through Lipinski's Rule of Five (property-oriented) and False Positive Remover (substructure-oriented), although many other criterions can be established.

Compound Filtering by Lipinski's Rule of Five

Lipinski's Rule of Five characterizes molecular properties important for oral bioavailability. In order to use Rule of Five as profiling tool, molecular properties, including molecular weight, cLogP, number of hydrogen bond donor, and number of hydrogen bond acceptor, were calculated by Sybyl for the compounds in subsets NDL-B and APL-C. The distribution of these properties is displayed in Figure S1. A filter was further created to remove compounds that meet any of the following criterions:

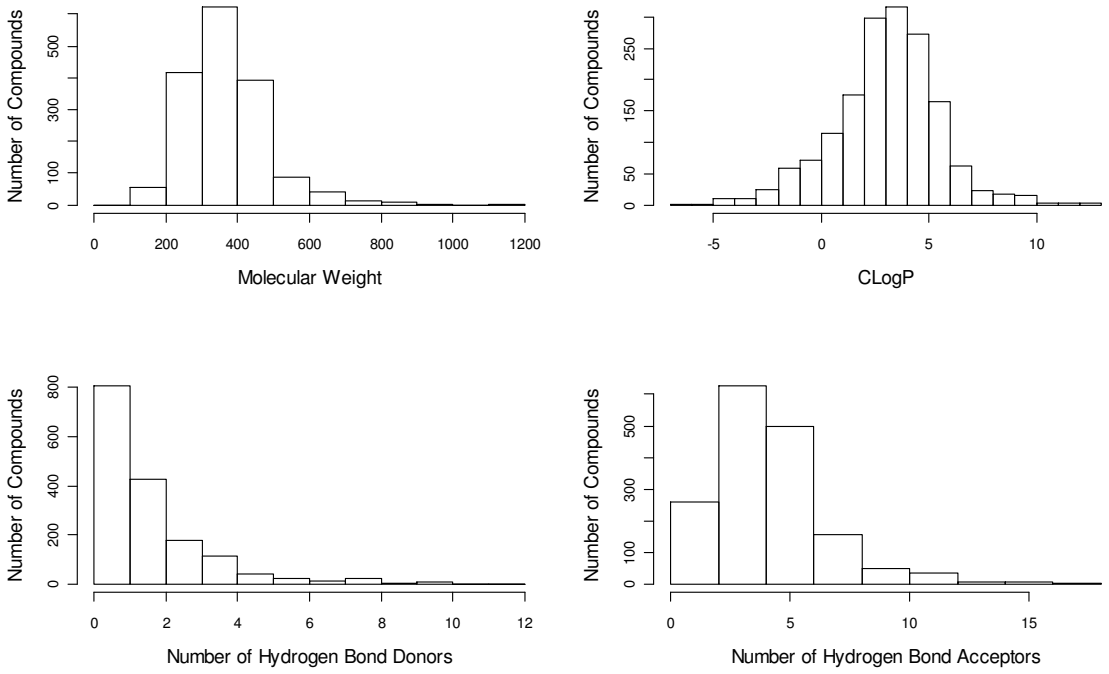
molecular weight > 500;
cLogP > 5;
the number of hydrogen bond donors > 5;
the number of hydrogen bond acceptors > 10.

Finally, 1230 compounds in the subset NDL-B and 997 compounds in the subset APL-C satisfied the Lipinski's Rule of Five.

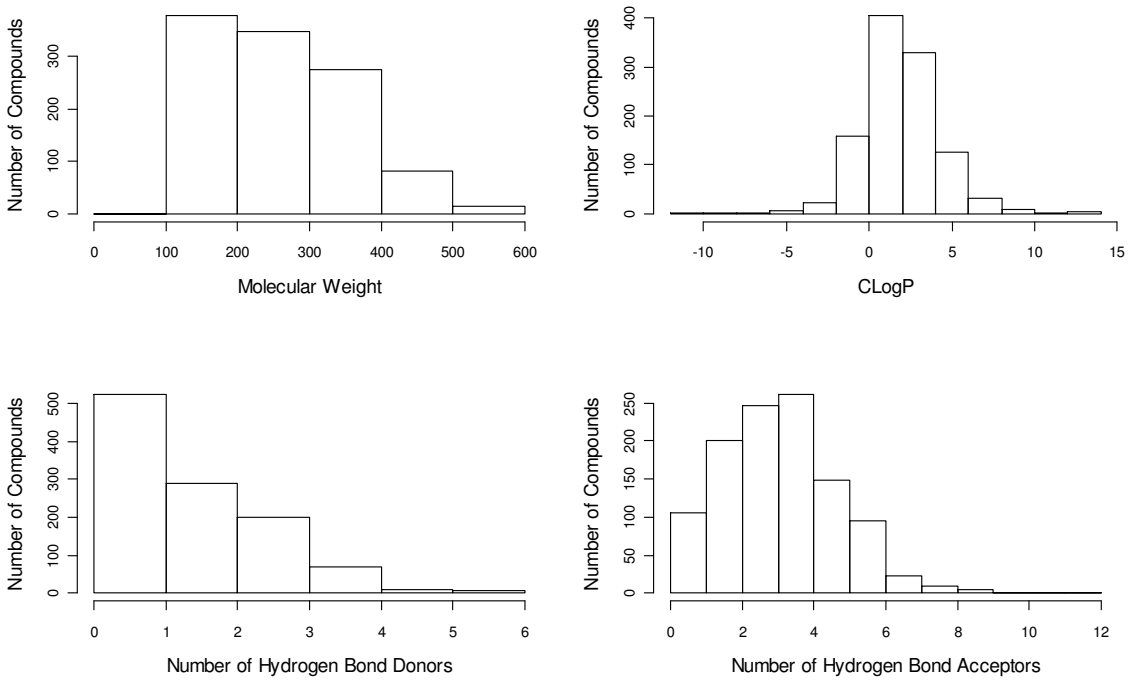
Compound Filtering by False Positive Remover

Different from Lipinski's Rule of Five, False Positive Remover aims at identifying compounds that interfere in high-throughput screening technology and trigger false positive signals. Such compounds are defined as Pan Assay Interference Compounds (PAINS).[1] PAINS can be identified by a list of signature substructures. This strategy is also applicable to searching compounds with toxicity groups. The methodology reported by Baell *et al*. was implemented as an online website "False Positive Remover", which is accessible at http://cbligand.org/PAINS/. 168 out of 1648 compounds in subset NDL-B were recognized as PAINS, and 85 out of 1096 compounds in subset APL-C were labeled as PAINS.

1. Baell, J. B.; Holloway, G. A. New Substructure Filters for Removal of Pan Assay Interference Compounds (PAINS) from Screening Libraries and for Their Exclusion in Bioassays. *J Med Chem* **2010,** *53* (7), 2719-2740.



(A)



(B)

Figure S1. The distribution of molecular properties, including molecular weight, cLogP, number of hydrogen bond donors, and number of hydrogen bond acceptors. Pane A is for the compounds in subset NDL-B, and pane B is for the compounds in subset APL-C.