ABSTRACT

We present gCOMBINE, a Java-written graphical user interface (GUI) for performing comparative binding energy (COMBINE) analysis (Ortiz et al. J Med Chem 1995; 38:2681–2691) on a set of ligand-receptor complexes with the aim of deriving highly informative quantitative structure-activity relationships. The essence of the method is to decompose the ligand-receptor interaction energies into a series of terms, explore the origins of the variance within the set using Principal Component Analysis, and then assign weights to selected ligand-residue interactions using partial least squares analysis to correlate with the experimental activities or binding affinities. The GUI allows plenty of interactivity and provides multiple plots representing the energy descriptors entering the analysis, scores, loadings, experimental versus predicted regression lines, and the evolution of parameters such as $r^2$ (correlation coefficient), $q^2$ (cross-validated $r^2$), and prediction errors as the number of extracted latent variables increases. Other representative features include the implementation of a sigmoidal dielectric function for electrostatic energy calculations, alternative cross-validation procedures (leave-N-out and random groups), drawing of confidence ellipses, and the possibility to carry out several additional tasks such as optional truncation of positive interaction energy values and generation of ready-to-use PDB files containing information related to the importance for activity of individual protein residues. This information can be displayed and color-coded using a standard molecular graphics program such as PyMOL. It is expected that this user-friendly tool will expand the applicability of the COMBINE analysis method and encourage more groups to use it in their drug design research programs.

INTRODUCTION

Accurate prediction of biological activities for newly designed molecules is one of the greatest challenges faced in computer-aided drug research. Because the number of three-dimensional structures for pharmacologically relevant targets is continually increasing, the pioneering quantitative structure-activity relationship (QSAR) methodologies that rely solely on physico-chemical parameters of substituents in congeneric series of compounds or on molecular interaction fields (MIF) calculated at discrete points in a three-dimensional (3D) lattice embedding the spatially aligned compounds, as in the popular comparative molecular field analysis (CoMFA) and comparative molecular similarity indices analysis (CoMSIA), have given way to other computational methods that attempt to derive as much information as possible from the structures of the ligand-receptor complexes. In this regard, continuous advances in the fields of comparative (homology) modeling of proteins of unknown experimental structure and automated ligand docking, as well as in computer power, have made it possible to search for new putative ligands for many different targets out of pools containing millions of candidate compounds (chemical libraries). Although this "virtual screening" approach has met with some success the technique is far from being mature because of several reasons, the most important one possibly being that the scoring functions used to rank the molecules and prioritize the possible hits are not accurate enough. When the number of compounds for a given target is kept under a few hundred, however, and biological activities are known, as is usually the case in a typical medicinal chemistry project, it...
should be feasible to employ a highly precise energy function to describe the ligand-receptor interactions and then derive an accurate QSAR with predictive ability using a method such as comparative binding energy (COMBINE) analysis.\textsuperscript{7}

COMBINE employs a number of residue-based interaction energies (both van der Waals and electrostatic) computed on a set of refined ligand-receptor complexes (rather than MIF calculated on a 3D grid for a set of unbound superimposed molecules, as is done in CoMFA) to build a data matrix that is then subjected to multivariate statistical analysis. The key idea is that partial least squares (PLS) can be used to correlate the computed energy components (plus additional optional terms such as receptor and ligand desolvation energies) with the experimental activities using an expression of the form shown in Eq. (1):

\[
\Delta G = \sum_{i=1}^{2n} w_i u_i + C
\]

where, for \( n \) protein residues (or protein residues plus selected water molecules), there are \( 2n \) terms, \( u_i \), each representing either a van de Waals or an electrostatic residue-based interaction energy with the ligand that contributes to the total binding free energy according to a weighting factor, \( w_i \), that is determined from the PLS analysis, and \( C \) is a constant. Although other partitioning schemes are also possible,\textsuperscript{7,8} only whole protein residues and ligands are supported in the present implementation. Obviously, if one of these contributions does not vary significantly among the complexes it cannot be used to account for activity/affinity differences within the series however important it may be for the constant term, \( C \), and the overall free energy of binding. A pictorial representation of the COMBINE workflow is shown in Figure 1.

Since its inception in 1995, when the original approach was developed to account for the differences in activity in a series of human synovial fluid phospholipid \( \Delta \) \( \varepsilon \) inhibitors,\textsuperscript{7} the COMBINE method has been applied to the study of other small molecules binding to different protein targets (HIV-1 protease,\textsuperscript{9} human cytochrome P450 1A2,\textsuperscript{10} human neutrophil elastase,\textsuperscript{11} HIV-1 reverse transcriptase (RT),\textsuperscript{12} acetylcholinesterase,\textsuperscript{13-15} haloalkane dehalogenases DhalA\textsuperscript{16} and LinB\textsuperscript{17}) and also to peptide-protein,\textsuperscript{18,19} protein-protein,\textsuperscript{20} and protein-DNA interactions.\textsuperscript{8} The method has acquired some relevance within the 3D-QSAR field and has been reviewed in detail by Wade et al.,\textsuperscript{21,22} Damborsky et al.,\textsuperscript{17} and more recently by Lushington et al.,\textsuperscript{23} who have also proposed some ideas to enhance COMBINE’s capabilities in the future.

An important milestone in the development of the COMBINE methodology was the incorporation of multiple structures into the analysis, which allows the introduction, at least in part, of target flexibility.\textsuperscript{8,24-26} All of these studies demonstrate that qualitatively reliable COMBINE models can be obtained using multiple structural representations of the receptor, but care must be taken when attempting to perform a quantitative analysis, due to the conformational dependence of the models. Another interesting issue related to structural variation entering a COMBINE analysis is the joint study of affinity and selectivity by use of different protein targets belonging to the same family, which may provide important guidelines for drug design. Key examples are the study by Wang and Wade\textsuperscript{27} of sialic and benzoic acid analogs binding to N2 and N9 subtypes of neuraminidase and the study of ligand binding to three serine proteases (trypsin, thrombin, and coagulation factor Xa) by Murcia et al.\textsuperscript{28} This approach can, in principle, be extended to an arbitrary number of receptors from the same protein family.

COMBINE analysis can also be linked to a docking algorithm, as shown by Murcia and Ortiz,\textsuperscript{25} when screening virtual libraries to derive more reliable bound conformations of the putative ligands, improve the predictive ability of the regression models, and increase the enrichment factors.

It can be seen therefore that COMBINE analysis occupies a privileged position as an effective tool that can aid in the design and optimization of drug candidates. However, as recently stated by Lushington et al.,\textsuperscript{23} the small number of groups worldwide using COMBINE should join their efforts to disseminate the method and make it available to the scientific community. Although our group promoted this initiative some time ago through the free Web-based release of the COMBINE code,\textsuperscript{29} a user-friendly graphical interface that allows easy manipulation of data and input/output files was lacking. In this contribution we fill this gap by providing the COMBINE

Figure 1  COMBINE workflow.
MATERIALS AND METHODS

Technical details of the gCOMBINE application

gCOMBINE is the GUI developed as a user-friendly wrapper to the original command-line COMBINE program. The GUI has been written in Java\(^{30}\) language (v. 1.6.0_10), which ensures platform portability. Graphics functionality and interactivity have been added with Java Foundation Classes (JFC) and Swing components, respectively. For development of the GUI the NetBeans\(^{31}\) IDE (Integrated Development Environment) 6.1 was used and the Swing Application Framework\(^{32}\) was included. The different charts were generated using the JFreeChart\(^{33}\) 1.0.11 and JCommon\(^{34}\) 1.0.14 libraries. Both are distributed under a GNU Lesser General Public License.\(^{35}\) These libraries allow gCOMBINE to generate interactive charts with the most relevant data for easy manipulation and analysis. Because the GUI is platform-independent and the COMBINE program is written in standard GNU Fortran, the complete application (COMBINE + gCOMBINE) can be used under Linux, Windows or Mac operating systems with the gcc compiler (v. 3.4.6 was employed in our case). gCOMBINE has an object-oriented design based on the Model-View-Controller (MVC) pattern.\(^{36}\) The main class for the Model is CombineModel. An instance of this class stores the information about a specific model (or configuration) generated from a COMBINE run: name for the model, a description comment, the working folder, the configuration parameters, the output files, and tables and charts. The parameters are stored on an instance of the Parameters class, which uses the ComplexesListItem objects to keep the different ligand-receptor complexes related to the COMBINE model under study. The tables and charts are panels generated by the static methods of the classes CombineTables and CombineGraphs, respectively, taking a CombineModel instance as input. The View is launched by the CombineGUIApp class that creates an instance of the CombineGUIView class. This instance acts as a store for the different graphical objects and also as the Controller for the different actions (including internal validations) that can be performed when interacting with the objects. An internal class (CombineThread) is used to run the COMBINE program in a different execution thread to avoid the blockade of the GUI interface while COMBINE is running. CombineThread uses an instance of the CombineWrapper class to prepare the COMBINE execution: it launches the calculation, controls the process (taking the logs with the StreamGlobber class) and loads the results upon completion of the run. Three other classes are used through the life cycle of the application: a) CombineConstants (to contain different constants); b) CombineException (to propagate customized errors and warnings); and c) Useful (to store some common methods).

The statistics behind COMBINE

As in any other method focused on obtaining a quantitative view of structure-activity relationships, the core of COMBINE is a matrix containing structure-related energy descriptors (variables) to be correlated with biological activities or binding energies. The statistical method underlying COMBINE analysis is well known and widely accepted by the scientific community, and therefore it has been intensely reviewed.\(^{37}\) However, a brief summary of the main steps and ideas follows.

Construction of the \(\mathbf{X}\) matrix

The \(\mathbf{X}\) matrix contains the entire set of variables describing the interaction energies between each ligand and every protein residue for all the complexes. Usually these are van der Waals interactions calculated using a molecular mechanics force field (typically AMBER) and electrostatic interactions calculated using point charges and either Coulomb’s law (and a constant or distance-dependent dielectric definition) or the more elaborate and accurate generalized born (GB)\(^{38}\) or Poisson-Boltzmann (PB)\(^{39}\) methods. In addition, desolvation energy terms for both receptor and ligand can also be incorporated as “external variables”. For each complex only two AMBER-type files are required, one containing the atomic coordinates (.crd extension) and the other containing the topology (i.e. atom connectivity), atom types and force-field parameters (.top extension). Alternatively, the user can generate the \(\mathbf{X}\) matrix externally in the format described in the User Guide and load it into the program.

Pretreatment of the \(\mathbf{X}\) matrix

To reduce the number of variables while keeping all the relevant information within the \(\mathbf{X}\) matrix, those interaction energy values with a standard deviation below a user-defined cut-off, which can be safely assumed not to contribute to the overall variance in activity, can be removed (Pretreatment cut-off, see below). Positive energy values, which in some cases could arise from force-field inconsistencies or modeling errors, can optionally be truncated to zero (Pretreatment option, see below). Scaling of the variables can also be performed using two different approaches (Scaling option, see below): (i) standard scaling, where the mean value over the whole set of variables is subtracted from each variable and divided by the standard deviation (it is therefore similar to a Z-score), and (ii) block scaling, by means of
which the mean value of the variables is subtracted from the one being scaled and divided by the standard deviation of these variables (again, similar to a Z-score but using a group of variables).

**PLS regression**

This technique combines and generalizes features from PCA and Multiple Linear Regression (MLR) in the sense that not only orthogonal Principal Components (PC) are extracted, as in PCA, but also a fitting procedure is performed to describe the activities of the compounds (the dependent variable), as in MLR. There are two initial matrices in a COMBINE analysis: (i) a matrix containing the independent variables (interaction energies, and possibly additional variables such as desolvation energies), the X matrix [Eq. (2)], and (ii) a matrix (column vector) with the dependent variable (activities), the Y matrix [Eq. (3)].

$$
X = \begin{bmatrix} E_{11} & E_{12} & \ldots & E_{1j} & V_{11} & V_{12} & \ldots & V_{1j} & A_{11} & A_{12} & \ldots & A_{1j} \\ E_{21} & E_{22} & \ldots & E_{2j} & V_{21} & V_{22} & \ldots & V_{2j} & A_{21} & A_{22} & \ldots & A_{2j} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ E_{N1} & E_{N2} & \ldots & E_{Nj} & V_{N1} & V_{N2} & \ldots & V_{Nj} & A_{N1} & A_{N2} & \ldots & A_{Nj} \end{bmatrix} \quad (2)
$$

where $E_{ij}$, $V_{ij}$, and $A_{ij}$ are the electrostatic, van der Waals, and additional variables, respectively. $N$ is the number of compounds, $M$ is the number of residues in the protein, and $S$ is the number of additional variables.

$$
Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \quad (3)
$$

where $y_i$ is the individual activity of compound $i$. The PLS analysis starts by decomposing the X and Y matrices into one score matrix, $T$, and two different loading matrices, $P$ and $Q$ [Eq. (4)], using the iterative NIPALS algorithm:

$$
X = TP^T \quad Y = TQ^T \quad (4)
$$

The loading matrices $P$ and $Q$ contain information about the variables in the so-called LV or PC space. These are orthogonal vectors obtained as linear combinations of the original variables in the X matrix. The coefficients in a given PC provide information on the relative weight of the different terms and can be used to deduce the relevance of each individual ligand–residue interaction to explain the variance in activity/affinity. On the other hand, the score matrix $T$ contains information about the compounds, described in terms of their projections onto the PCs. The PC space is normalized and has a mean of zero, so compounds with high scores should be checked as they could behave as outliers. In addition, clusters of compounds can be detected. A plot of the regression line between the experimentally determined and theoretically calculated activity/affinity values and calculation of the regression coefficient [$r^2$, Eq. (5)] allow the user to visualize the quality of the fit for the training set compounds, and also for the excluded (not used) or test compounds.

$$
\begin{align*}
    r^2 &= \frac{\sum_{i=1}^{N} (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})^2}{\sum_{i=1}^{N} (y_i - \bar{y})^2 \sum_{i=1}^{N} (\hat{y}_i - \bar{\hat{y}})^2} \\
    \bar{y} &= \frac{\sum_{i=1}^{N} y_i}{N}
\end{align*} \quad (5)
$$

where $\bar{y}$ is the average value of the activity ($\bar{y} = \sum_{i=1}^{N} y_i / N$). In simple words, this metric describes the amount of variance in the dataset that is explained by the model. Besides, a standard deviation of error of predictions [SDep, Eq. (7)] and an average absolute error [AAE, Eq. (8)] are also calculated.

\[\text{AAE} = \text{SDep} \text{Eq. (7)}\]
It can be argued that, despite the cross-validation procedure, the resulting model fits the data just by chance due to the selection of a fortuitous equation out of the huge amount of different PLS regression models that can be constructed with the thousands of variables contained in the X matrix. To check against this possibility the affinities/activities of the compounds can be randomly reassigned (permutation of activities or Y-randomization) to prove the point that in this case it is usually not possible to derive an acceptable model. gCOMBINE allows the user to carry out this task thorough the Y-randomization option in the main window (see below). This test is performed 100 times and therefore it is quite time-consuming.

**Selection of the best model**

As successive components are extracted from the X matrix, a check is made to estimate the amount of variance that is recovered (it must be borne in mind that the PLS method attempts to explain the variance not only in the X matrix, as does PCA, but also in the Y matrix). Although there is not a strict rule to select the best model resulting from a PLS analysis, the general guidance is to study the evolution of both the cross-validated correlation coefficient [$q^2$, Eq. (6)] and the standard deviation of the errors in prediction [SDEP, Eq. (7)]. gCOMBINE provides the user with a graphical description showing the evolution of the main chemometric indices as new components are being extracted (five by default, and up to 10, Number of Latent Variables option, see later) to facilitate the decision on the optimal dimensionality to choose for further analysis.

**External validation**

The best way to validate a PLS model is to challenge it with an external set of modeled complexes and compare the predicted affinity/activity values for the bound complexes.
ligands with the actual ones. This is accomplished in gCOMBINE by feeding the program with additional complexes that are marked as “Test” in lieu of “Training” in the Type column (see Main Menu in Fig. 2).

RESULTS AND DISCUSSION

The application

gCOMBINE has been designed bearing in mind the need for a simple and easy-to-use chemometric tool. Once that the program has been launched it displays a menu bar with two submenus (File and Help) and a tabbed panel with two tabs (see Fig. 2): one for the configuration of the model (COMBINE Model) and the other one to manage the results (Results). The menu bar, under File submenu, contains four options: (a) New Model, to clean the data from the tabs, (b) Load Model, to load a previously saved model into the tabs, (c) Save Model, to save the current model into a specific file, and (d) Exit, to close the application. Help submenu offers information about the original COMBINE publication, the main COMBINE author and contact information. The COMBINE Model tab can be divided into four main areas (a through d in Fig. 2): (a) the top part where the user can select the folder for the COMBINE executable and the working folder where the complexes are stored. Clicking on the RUN COMBINE button will start the calculation, (b) a section where the user can introduce commentaries related to the job into two boxes, Name and Description, (c) this section allows the user to load parameters from a previous calculation or to save the current parameters being used thorough the Load/Save Parameters buttons. All the parameters entering the model are configured here (the reader is referred to the “Materials and Methods” section for the definition of the different issues described here): Y-randomization (No by default), Scaling (No by default), Interaction Matrix (it can be calculated and written out by gCOMBINE or read in from an external file), Number of Latent Variables (5 are extracted by default), validation method (Leave N Out or Random Groups), Type of Electrostatic Model (uniform dielectric constant, Goodford’s implementation of the images method, a distance-dependent dielectric constant, PB electrostatic interaction energies read from an external file, and a sigmoidal model), Dielectric...
Constant value (four by default), Number of External Variables, Pretreatment of the data and the Pretreatment cut-off value, and (d) the complex area, where the user can add/remove/load/save complexes from/to a file (Add/Remove Complex and Load/Save from/to File buttons). Below these buttons a table is shown with the following information: the type associated to each complex (Training, Test, or Not Used, as defined by the user), which can be changed at any time to test alternative models, the File Name containing the complex, the Ligand Name, the Pharmacological Activity, and one column for each external variable to be considered. Once the parameters and the complexes have been read in and after the program begins to run, the application checks if all the parameters needed have been supplied, if they have valid values, and if the required files exist.

The user can stop the execution or wait until completion of the run. In this latter case, if no errors are found, the application focuses on the Results tab (see Fig. 3) where the user can have access to different tables and charts grouped by nodes and sub-nodes: a summary of each model (both a graphic showing the evolution of the chemometric indices and a table containing these indices are produced), PLS coefficients plots (Fig. 4), predictions (for each model a graphic of experimental vs. predicted activity values is shown, Fig. 5 panel a), loadings and scores plots (Fig. 5 panels b and c, respectively), and a plot of the interaction energy variables entering the PLS decomposed on a per-residue basis (a very useful plot to detect anomalous energy values, Fig. 5 panel d). The user can interact with the graphs in several ways: zoom in/out, see tooltips for specific data, set tags, change the appearance, save them as images, print, and so forth. Finally, there is a Logs node to keep track of the program messages.

Testing gCOMBINE

Among the many publications of successful COMBINE analysis, we have chosen two of them for testing the graphical implementation reported herein. The first one is the set accompanying the original distribution, which employed 48 (32 for the training set and 16 for the test set) inhibitors of human immunodeficiency virus Type 1 protease.
In the second example, also related to another pharmacologically relevant HIV-1 target, namely RT, we will show how externally generated electrostatic energy variables, as calculated with the DelPhi software, can be incorporated into the model.

1. Two main issues motivated the first of these two studies: (a) bare ligand-receptor interaction energies (as computed by Merck researchers using the MM2X force field) per se correlated quite well (using linear regression analysis, \( r^2 = 0.74 \)) with experimentally determined enzyme inhibition data (IC_{50} values, that is, compound concentrations giving rise to 50% inhibition of enzyme activity). Moreover, the predictive ability of such a linear model on 16 test set compounds (not included in model derivation) was also remarkable (\( q^2 = 0.75 \)) with an average absolute error around 1 across a range of 5 log units, and (b) the realization that no improvement upon incorporation of either solvation effects (using a continuum description) or using another force field (CHARMM in their case) was achieved. Taking these two issues into account, the two main objectives addressed by the

<table>
<thead>
<tr>
<th>Model</th>
<th>Objects</th>
<th>Variables</th>
<th>LV</th>
<th>( r^2 )</th>
<th>( q^2 )</th>
<th>SDEP_Cv</th>
<th>SDEP_MK</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMM2X</td>
<td>32</td>
<td>1</td>
<td>1</td>
<td>0.74</td>
<td>0.75</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>LAMBER</td>
<td>32</td>
<td>1</td>
<td>1</td>
<td>0.81</td>
<td>0.79</td>
<td>0.61</td>
<td>1.08</td>
</tr>
<tr>
<td>CAMBER</td>
<td>32</td>
<td>48</td>
<td>2</td>
<td>0.89</td>
<td>0.70</td>
<td>0.72</td>
<td>0.83</td>
</tr>
<tr>
<td>gCAMBER</td>
<td>32</td>
<td>48</td>
<td>2</td>
<td>0.89</td>
<td>0.70</td>
<td>0.72</td>
<td>0.80</td>
</tr>
</tbody>
</table>

*Calculated with gCOMBINE.
COMBINE analysis methodology were (i) to check for possible dependencies of the correlation on the force field used, and (ii) to try and develop more accurate QSAR models. In this exercise, it was shown that similar results could be obtained when comparable models were built within the framework of the AMBER force field, so no force-field dependencies were detected (see the chemometric indices for L_{MM2X} and L_{AMBER} in Table I). On the other hand, remarkable improvements were achieved through the use of COMBINE models (see the original article), especially when partial desolvation effects for ligand and receptor upon complex formation were included using a continuum description (by solving the PB equation) and the standard Coulombic distance-dependent electrostatic term was replaced with solvent-corrected values calculated for each residue. Interestingly, the main conclusion of this work was that simply replacing the Coulombic term with the continuum electrostatics description and including the desolvation effects did not lead to a significant improvement when MLR was used but the performance of the corresponding COMBINE model was dramatically enhanced when the continuum electrostatic interactions were employed. Although different improvements over the standard Coulombic term were included in the original article we will restrict ourselves here to the simplest (and more widely used) case as our intention is to reproduce the data rather than recapitulating the previously published comparison. In particular we show the reproducibility of what was called the C_{AMBER} model, where van der Waals and electrostatic contributions (the latter, a straightforward Coulombic term using a dielectric constant of four) were taken directly from the AMBER force field using the ANAL module. The results cannot be exactly the same, though, because the cross-validation technique contains a random component: the compounds are randomly assigned to one of five groups of approximately the same size, each group in turn is excluded from the analysis, and the whole procedure is repeated 20 times. Nevertheless, clearly comparable results were obtained (see chemometric indices for C_{AMBER} and gC_{AMBER} in Table I).

Finally, the relative weight assigned to individual residue-based interactions (van der Waals and electrostatic) by the COMBINE model can be color-coded and displayed on a surface representation of the protein, as shown in Figure 6.

2. The second article revisited here entailed the study of 27 6-arylsulfonyl-2-aminobenzonitrile derivatives synthesized and tested as second-generation non-nucleoside HIV-1 RT inhibitors (NNRTI). In this case there was a wealth of experimental data including information about activity on RT enzymes bearing different mutations at the NNRTI binding site. COMBINE models were obtained to quantitatively characterize the observed structure-activity data and possibly to account for the effect of some of the mutations. Ligand-residue van der Waals interaction energies were calculated using AMBER parameters (parm99) while their solvent-corrected electrostatic counterparts were obtained by solving the PB equa-


Table II
Chemometric Indices Calculated by gCOMBINE for the Whole Set of 27 6-aryl-sulfonyl-2-aminobenzonitrile HIV-1 Reverse Transcriptase Inhibitors

<table>
<thead>
<tr>
<th>LV</th>
<th>$r^2$</th>
<th>$q^2$</th>
<th>SDEP$_{CV}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.84</td>
<td>0.71</td>
<td>0.62</td>
</tr>
<tr>
<td>2</td>
<td>0.88</td>
<td>0.78</td>
<td>0.55</td>
</tr>
<tr>
<td>3</td>
<td>0.93</td>
<td>0.82</td>
<td>0.49</td>
</tr>
<tr>
<td>4</td>
<td>0.94</td>
<td>0.86</td>
<td>0.44</td>
</tr>
<tr>
<td>5</td>
<td>0.95</td>
<td>0.86</td>
<td>0.44</td>
</tr>
</tbody>
</table>

The objective of this article has been to provide the COMBINE analysis method with an easy-to-use GUI that improves on the original command-line style implementation. The software is written in Java to allow platform portability and is made freely available to academic and/or public research institutions from a public web site (http://ub.cbm.uam.es/gCOMBINE) under an Academic License. This has been done with the idea of disseminating a user-friendly tool among the scientific community to encourage the use of a program that has proven useful in many areas related to ligand binding, structure-activity relationships and drug design.

ACKNOWLEDGMENTS

The authors thank Claire Coderch for help in testing the application. This work is dedicated to the memory of Angel R. Ortiz, who looked forward to making this tool widely available but passed away before it was fully operative.

REFERENCES


