ACS OMEGA

Article

Cite This: ACS Omega 2017, 2, 8682–8688

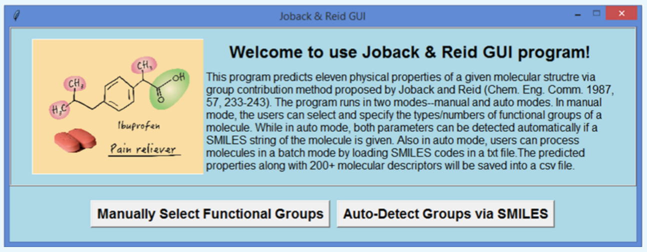
JRgui: A Python Program of Joback and Reid Method

Chenyang Shi* and Thomas B. Borchardt

Drug Product Development, AbbVie, 1 North Waukegan Road, North Chicago, Illinois 60064, United States

Supporting Information

ABSTRACT: Using the modern object-oriented programing language Python (e.g., *tkinter* and *pandas* modules) and a chemoinformatics open-source library (RDKit), the classic Joback and Reid group contribution method was revisited and written into a graphical user interface program, JRgui. The underlying algorithm behind the program is explained, herein, with the users being able to operate the program in either a manual or automatic mode. In the manual mode, the users are required to determine the type and occurrence of functional groups in the compound of interest and manually enter into the program. In the automatic mode, both of these parameters can be detected automatically via user input of the compound simplified molecular input line entry specification (SMILES) string. An additional advantage of the automatic mode is that a large number of molecules can be processed simultaneously by parsing their individual SMILES strings into a text file, which is read by the program. The resulting predicted physical properties along with approximately 200 molecular descriptors are saved in a spreadsheet file for subsequent analysis. The program is available for free at https://github.com/curieshicy/JRgui for Windows, Linux, and macOS 64-bit operating systems. It is hoped that the current work may facilitate the creation of other user-friendly programs in the chemoinformatics community using Python.


Joback & Reid GUI
CH3
OH
H3C
CH3
O
Ibuprofen
Pain reliever
Manually Select Functional Groups
Auto-Detect Groups via SMILES

Welcome to use Joback & Reid GUI program!

This program predicts eleven physical properties of a given molecular structre via group contribution method proposed by Joback and Reid (Chem. Eng. Comm. 1987, 57, 233-243). The program runs in two modes--manual and auto modes. In manual mode, the users can select and specify the types/numbers of functional groups of a molecule. While in auto mode, both parameters can be detected automatically if a SMILES string of the molecule is given. Also in auto mode, users can process molecules in a batch mode by loading SMILES codes in a txt file.The predicted properties along with 200+ molecular descriptors will be saved into a csv file.

INTRODUCTION

In the early drug-screening stage, a quick estimation of the physical properties of proposed drug molecules is highly desired as it may save a large amount of resource in optimization of chemical matter and subsequent dosage form development. Of all of the available methods, the prediction of pure-component properties from group contributions (GCs) is an easy and straightforward method. The GC method is based on the principle that the physical properties of a molecule have contributions from the individual functional groups or atomic properties within a molecule. A general formula for the GC method is provided in eq 1

$$f = c + \sum_i n_i \Delta f_i \tag{1}$$

where n_i is the occurrence of ith group in the structure, Δf_i is the contribution to the properties from ith group, and c is a constant. The GC method has been developed from pure additive group contributions to including second-order or higher-order interactions from the groups.[1−7]

In the past few decades, a number of GC methods have been developed, and some of them are highly cited in the scientific community.[1−7] One of the highly cited methods has been provided by Joback and Reid (JR method, >1100 citations based on Google Scholar).[2] The JR method divides the common functional groups into six categories, including ring, nonring, halogen, oxygen, nitrogen, and sulfur increments. On the basis of the types and occurrence of the functional groups, the JR method predicts 11 physical properties, including normal boiling point, normal freezing point, critical temperature, critical pressure, critical volume, enthalpy of formation, Gibbs energy of formation, heat capacity, enthalpy of vaporization, enthalpy of fusion, and liquid viscosity.

To develop GC programs, computers have to be able to recognize chemical structures/functional groups, which can be achieved by storing them as molecular graphs whose edges and nodes represent, respectively, bonds and atoms. Alternatively, one may use a connection table or line notation to communicate the structural information to computers.[8] A connection table contains lists of the atomic number of atoms and atomic bonds, with additional information, such as hybridization state and bond order. Line notation, on the other hand, uses alphanumeric characters to represent key characteristics of the two-dimensional structure of a molecule. It contains precise information on the molecular fragments and their connectivity. Of its typical formats, simplified molecular input line entry specification (SMILES) and SMILES arbitrary target specification (SMARTS) are widely used. Developed by daylight chemical information system,[9] SMILES accurately describes a plethora of chemical information of a molecule, such as aromatic rings, number of hydrogen atoms, and atom-to-atom mapping. SMARTS was developed based on SMILES, specifically for describing the molecular fragment.

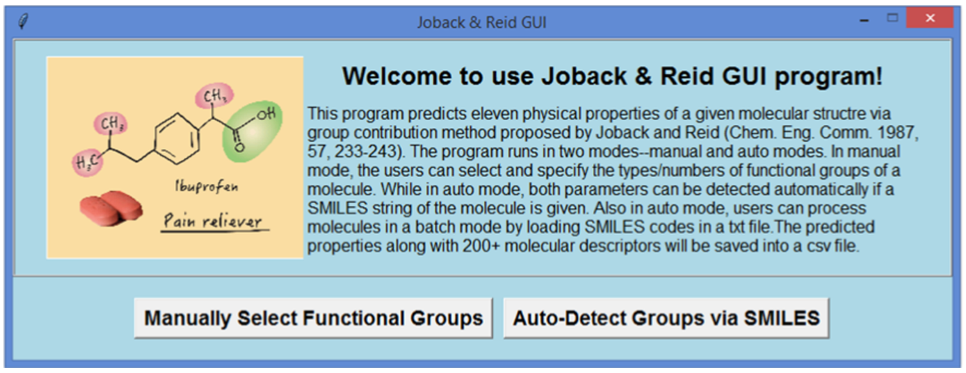
There are several online and commercial programs that have implemented the JR method.[10] However, the algorithm, in particular for automatch functional groups, is not published. In

Received: October 1, 2017
Accepted: November 16, 2017
Published: December 7, 2017

Table 1. Equations from the JR Method for Estimation of Physical Properties[a]

normal boiling point (K)	$T_b = 198.2 + \Sigma$ (2)
normal freezing point (K)	$T_f = 122.5 + \Sigma$ (3)
critical temperature (K)	$T_c = T_b/[0.584 + 0.965\Sigma - (\Sigma)^2]$ (4)
critical pressure (bar)	$P_c = (0.113 + 0.0032n_A - \Sigma)^{-2}$ (5)
critical volume (cm^3/mol)	$V_c = 17.5 + \Sigma$ (6)
enthalpy of formation (kJ/mol) at 298 K	$\Delta H^0_{f,298} = 68.29 + \Sigma$ (7)
Gibbs energy of formation (kJ/mol) at 298 K	$\Delta G^0_{f,298} = 53.88 + \Sigma$ (8)
heat capacity, ideal gas (J/mol K)	$C^0_p = \Sigma(a) - 37.93 + [\Sigma(b) + 0.210]T + [\Sigma(c) - 3.91 \times 10^{-4}]T^2 + [\Sigma(d) + 2.06 \times 10^{-7}]T^3$ (9)
enthalpy of vaporization at T_b (kJ/mol)	$\Delta H_{v,b} = 15.30 + \Sigma$ (10)
enthalpy of fusion (kJ/mol)	$\Delta H_f = -0.88 + \Sigma$ (11)
liquid viscosity (N s/m^2)	$\eta_L = \text{MW} \exp\{[\Sigma(\eta_A) - 597.82]/T + \Sigma(\eta_B) - 11.202\}$ (12)

[a]$\sum$ represents the group contributions. n_A and MW are the number of atoms and molecular weight, respectively.


Joback & Reid GUI

Welcome to use Joback & Reid GUI program!

This program predicts eleven physical properties of a given molecular structre via group contribution method proposed by Joback and Reid (Chem. Eng. Comm. 1987, 57, 233-243). The program runs in two modes--manual and auto modes. In manual mode, the users can select and specify the types/numbers of functional groups of a molecule. While in auto mode, both parameters can be detected automatically if a SMILES string of the molecule is given. Also in auto mode, users can process molecules in a batch mode by loading SMILES codes in a txt file.The predicted properties along with 200+ molecular descriptors will be saved into a csv file.

Ibuprofen
Pain reliever
Manually Select Functional Groups
Auto-Detect Groups via SMILES

Figure 1. Main window of JRgui program. The buttons at the bottom will link to manual and automatic modes, respectively.

addition, it is not straightforward to calculate the properties for a large number of molecules simultaneously when using a Web-based program.

In recent years, the Python programing language has been gaining increasing attention in the scientific community,[11] and several programs for chemoinformatics research are reported.[12−15] In this work, we describe the application of the modern object-oriented programing language Python, specifically the *tkinter* module,[16a] to create JRgui, a graphical user interface program based on the JR method. The program utilizes the *pandas* module within Python[16b] for a quick save of data, and the open-source chemoinformatics package, RDKit,[15] for input of requisite functional group descriptors. We demonstrate how to search subgroups using SMARTS code and how to remove duplicate functional groups in the search.

When using JRgui, the users can either manually select types/numbers of groups of molecules of interest or have the computer algorithm automatically input functional group information via input of a SMILES string for the molecule of interest. In the case of executing calculations on a series of molecules simultaneously (e.g., a batch process), the users are required to store all SMILES strings into a text file for subsequent input to the program. In automatic mode, the JRgui program will automatically save and export predicted properties along with approximately 200 molecular descriptors into a spreadsheet for subsequent evaluation.

EQUATIONS

In this section, the equations used in JRgui are summarized and explained. First, the original equations to calculate 11 physical properties by the JR method are listed in Table 1 (eqs 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12).

In the manual mode, an equation proposed by Mannhold et al. is used to predict log *P*

$$\log P = 1.46 + 0.11 \times N_C - 0.11 \times N_{HET} \qquad (13)$$

where N_C and N_{HET} stand for the number of carbon atoms and the number of heteroatoms, respectively.[17] The authors surveyed more than 96 000 compounds before they arrived at this empirical relation.

Because the JR method predicts the melting point of molecules, the general solubility equation[18] and its modified form[19] can be used to predict its crystalline solubility via

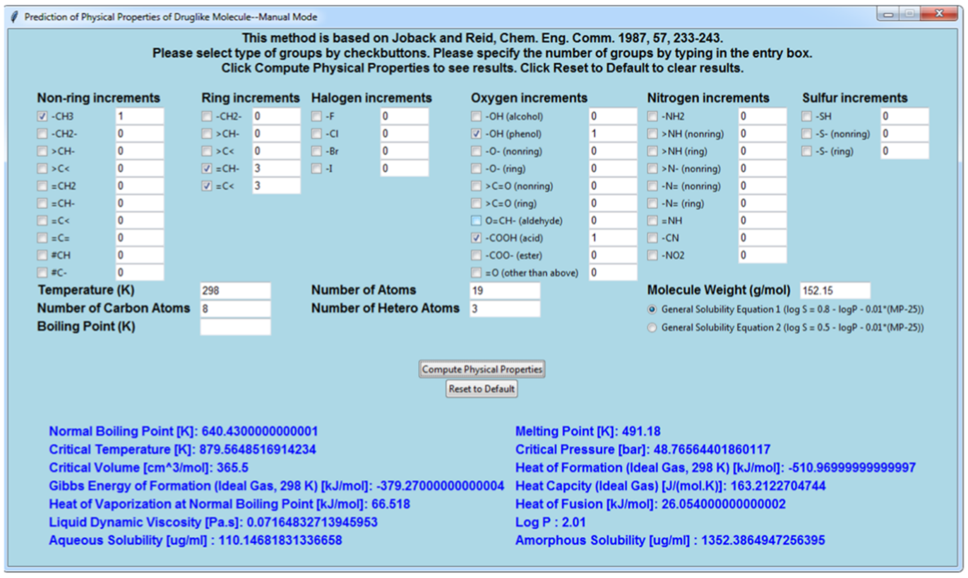
$$\log S = 0.8 - \log P - 0.01 \times (\text{MP} - 25) \qquad (14)$$

$$\log S = 0.5 - \log P - 0.01 \times (\text{MP} - 25) \qquad (15)$$

where *S* is the solubility in mol/L, log *P* is the lipophilicity, and MP is the melting point in °C.

The crystalline solubility can be converted to that in μg/mL by multiplying by molecular weight (in g/mol).

$$S = 10^{[0.8 - \log P - 0.01 \times (\text{MP} - 25)]} \times \text{MW} \times 1000 \qquad (16)$$



Prediction of Physical Properties of Druglike Molecule--Manual Mode

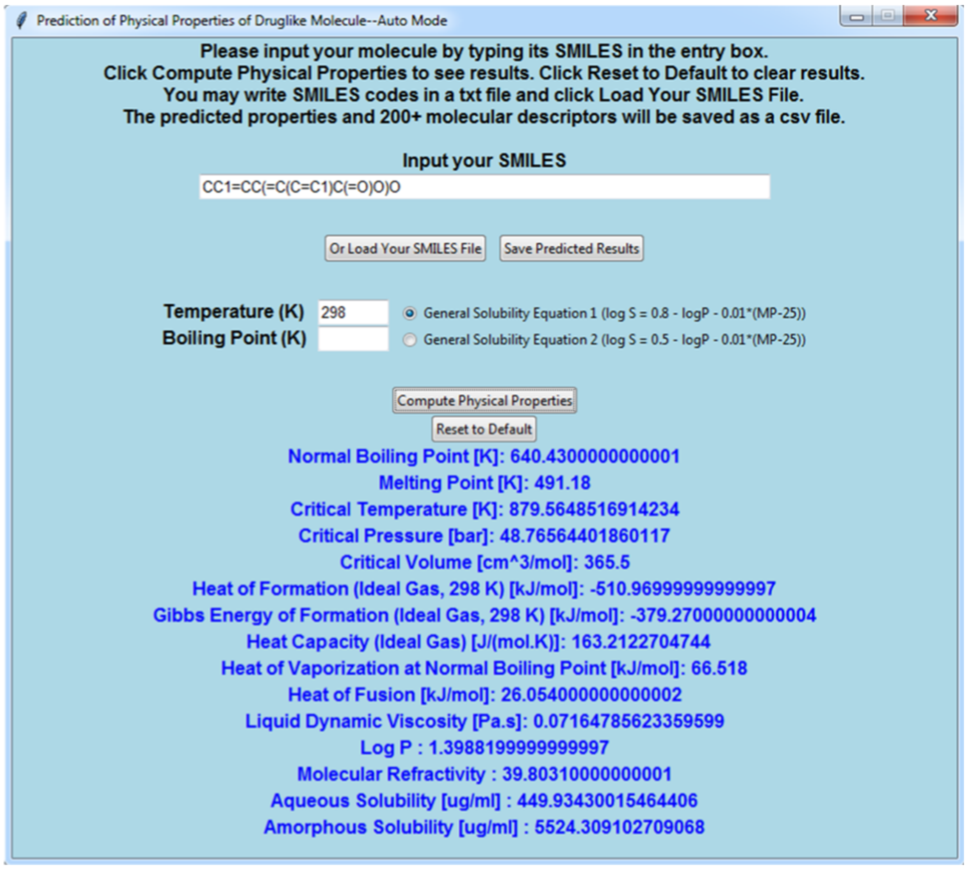
This method is based on Joback and Reid, Chem. Eng. Comm. 1987, 57, 233-243.
Please select type of groups by checkbuttons. Please specify the number of groups by typing in the entry box.
Click Compute Physical Properties to see results. Click Reset to Default to clear results.

Non-ring increments		Ring increments		Halogen increments		Oxygen increments		Nitrogen increments		Sulfur increments	
✓ -CH3	1	-CH2-	0	-F	0	-OH (alcohol)	0	-NH2	0	-SH	0
-CH2-	0	>CH-	0	-Cl	0	✓ -OH (phenol)	1	>NH (nonring)	0	-S- (nonring)	0
>CH-	0	>C<	0	-Br	0	-O- (nonring)	0	>NH (ring)	0	-S- (ring)	0
>C<	0	✓ =CH-	3	-I	0	-O- (ring)	0	>N- (nonring)	0		
=CH2	0	✓ =C<	3			>C=O (nonring)	0	-N= (nonring)	0		
=CH-	0					>C=O (ring)	0	-N= (ring)	0		
=C<	0					O=CH- (aldehyde)	0	=NH	0		
=C=	0					✓ -COOH (acid)	1	-CN	0		
#CH	0					-COO- (ester)	0	-NO2	0		
#C-	0					=O (other than above)	0				

Temperature (K) 298
Number of Carbon Atoms 8
Boiling Point (K)
Number of Atoms 19
Number of Hetero Atoms 3
Molecule Weight (g/mol) 152.15
(● General Solubility Equation 1 (log S = 0.8 - logP - 0.01*(MP-25)))
General Solubility Equation 2 (log S = 0.5 - logP - 0.01*(MP-25))

```
Normal Boiling Point [K]: 640.4300000000001
Critical Temperature [K]: 879.5648516914234
Critical Volume [cm^3/mol]: 365.5
Gibbs Energy of Formation (Ideal Gas, 298 K) [kJ/mol]: -379.27000000000004
Heat of Vaporization at Normal Boiling Point [kJ/mol]: 66.518
Liquid Dynamic Viscosity [Pa.s]: 0.07164832713945953
Aqueous Solubility [ug/ml] : 110.14681831336658
Melting Point [K]: 491.18
Critical Pressure [bar]: 48.76564401860117
Heat of Formation (Ideal Gas, 298 K) [kJ/mol]: -510.96999999999997
Heat Capcity (Ideal Gas) [J/(mol.K)]: 163.2122704744
Heat of Fusion [kJ/mol]: 26.054000000000002
Log P : 2.01
Amorphous Solubility [ug/ml] : 1352.3864947256395
```

Figure 2. Screenshot of JRgui program at manual mode.



Prediction of Physical Properties of Druglike Molecule--Auto Mode

**Please input your molecule by typing its SMILES in the entry box.
Click Compute Physical Properties to see results. Click Reset to Default to clear results.
You may write SMILES codes in a txt file and click Load Your SMILES File.
The predicted properties and 200+ molecular descriptors will be saved as a csv file.**

Input your SMILES
CC1=CC(=C(C=C1)C(=O)O)O

Temperature (K) 298
Boiling Point (K)
(● General Solubility Equation 1 (log S = 0.8 - logP - 0.01*(MP-25)))
General Solubility Equation 2 (log S = 0.5 - logP - 0.01*(MP-25))

```
Normal Boiling Point [K]: 640.4300000000001
Melting Point [K]: 491.18
Critical Temperature [K]: 879.5648516914234
Critical Pressure [bar]: 48.76564401860117
Critical Volume [cm^3/mol]: 365.5
Heat of Formation (Ideal Gas, 298 K) [kJ/mol]: -510.96999999999997
Gibbs Energy of Formation (Ideal Gas, 298 K) [kJ/mol]: -379.27000000000004
Heat Capacity (Ideal Gas) [J/(mol.K)]: 163.2122704744
Heat of Vaporization at Normal Boiling Point [kJ/mol]: 66.518
Heat of Fusion [kJ/mol]: 26.054000000000002
Liquid Dynamic Viscosity [Pa.s]: 0.07164785623359599
Log P : 1.3988199999999997
Molecular Refractivity : 39.80310000000001
Aqueous Solubility [ug/ml] : 449.93430015464406
Amorphous Solubility [ug/ml] : 5524.309102709068
```

Figure 3. Screenshot of the JRgui program at auto mode.

$$S = 10^{[0.5 - \log P - 0.01 \times (\mathrm{MP} - 25)]} \times \mathrm{MW} \times 1000 \tag{17}$$

Finally, the amorphous solubility advantage relative to that predicted for the crystalline phase can be estimated by the Gibbs free-energy difference in both states.[20]

$$\frac{S_a}{S_c} = e^{\Delta G / RT} \tag{18}$$

$$\Delta G = \frac{\Delta H_f (T_m - T) T}{T_m^2} \tag{19}$$

where S_a and S_c are the amorphous and crystalline solubilities of the compound, respectively, ΔH_f and T_m are, respectively, the heat of fusion and melting point of the drug molecule, which can be predicted by the JR method.

Table 2. Summary of 41 Functional Groups Used in the JR method and Their Corresponding SMARTS Codes Used in JRgui Program[9,15,22a]

nonring increments		ring increments		halogen increments	
$-CH_3$	[CX4H3]	$-CH_2-$	[R;CX4H2]	−F	[F]
$-CH_2-$	[!R;CX4H2]	>CH−	[R;CX4H]	−Cl	[Cl]
>CH−	[!R;CX4H]	>C<	[R;CX4H0]	−Br	[Br]
>C<	[!R;CX4H0]	═CH−	[R;CX3H1,cX3H1]	−I	[I]
$═CH_2$	[CX3H2]	═C<	[$([R;#6X3H0]);!$([R;#6X3H0]=[#8])]		
═CH−	[!R;CX3H1;!$([CX3H1](=O))]				
═C<	[$([!R;#6X3H0]);!$([!R;#6X3H0]=[#8])]				
═C═	[$([CX2H0](=*)=*)]				
#CH	[$([CX2H1]#[!#7])]				
#C−	[$([CX2H0]#[!#7])]				
oxygen increments		**nitrogen increments**		**sulfur increments**	
−OH (alcohol)	[OX2H;!$([OX2H]-[#6]=[O]);!$([OX2H]-a)]	$-NH_2$	[NX3H2]	−SH	[SX2H]
−OH (phenol)	[O;H1;$(O-!@c)]	>NH (nonring)	[NX3H1;!R]	−S− (nonring)	[#16X2H0;!R]
−O− (nonring)	[OX2H0;!R;!$([OX2H0]-[#6]=[#8])]	>NH (ring)	[#7X3H1;R]	−S− (ring)	[#16X2H0;R]
−O− (ring)	[#8X2H0;R;!$([#8X2H0]~[#6]=[#8])]	>N− (nonring)	[#7X3H0;!$([#7](~O)~O)]		
>C═O (nonring)	[$([CX3H0](=[OX1]));!$([CX3](=[OX1])-[OX2]);!R]=O	−N= (nonring)	[#7X2H0;!R]		
>C═O (ring)	[$([#6X3H0](=[OX1]));!$([#6X3](=[#8X1])~[#8X2]);R]=O	−N= (ring)	[#7X2H0;R]		
O#CH− (aldehyde)	[CH;D2;$(C-!@C)](=O)	═NH	[#7X2H1]		
−COOH (acid)	[OX2H]-[C]=O	−CN	[#6X2]#[#7X1H0]		
−COO− (ester)	[#6X3H0;!$([#6X3H0](~O)(~O)(~O))](=[#8X1])[#8X2H0]	$-NO_2$	[$([#7X3,#7X3+][!#8])](=[O])~[O-]		
═O (except as above)	[OX1H0;!$([OX1H0]~[#6X3]);!$([OX1H0]~[#7X3]~[#8])]				

[a]The codes are double-checked by visualizing them via online tool SMARTSviewer at Center for Bioinformatics at University of Hamburg.[23]

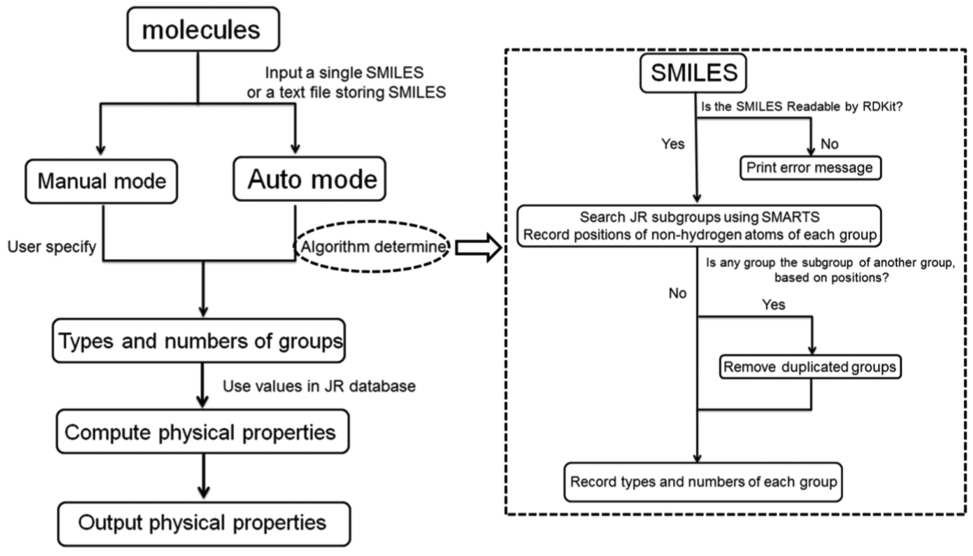

molecules
Input a single SMILES or a text file storing SMILES
Manual mode
Auto mode
User specify
Algorithm determine
Types and numbers of groups
Use values in JR database
Compute physical properties
Output physical properties
SMILES
Is the SMILES Readable by RDKit?
Yes
No
Print error message
Search JR subgroups using SMARTS
Record positions of non-hydrogen atoms of each group
Is any group the subgroup of another group, based on positions?
No
Yes
Remove duplicated groups
Record types and numbers of each group

Figure 4. Schematic diagram illustrating the workflow of JRgui program.

In automatic mode, more than 200 molecular descriptors are predicted using RDKit package, which can be used in further machine learning study. More details about these descriptors are documented in the RDKit manual. We note in the automatic mode that log P (and molecular refractivity) prediction is based on a different method by Wildman and Crippen.[21]

USAGE

The main window of the program is displayed in Figure 1. It has a brief description on top, and two buttons at the bottom, which link to the manual and automatic operation modes of the program, respectively.

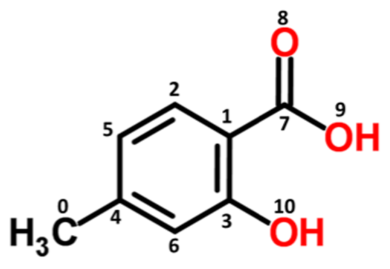
If the *Manually Select Functional Groups* button is clicked, the users will be directed to a pop-out window, as shown in Figure 2. The users can then manually select different types of groups via check buttons and enter the number of each group in the appropriate entry boxes. Some additional information is needed from users, which includes temperature, number of atoms, molecular weight, number of carbon atoms, and number of heteroatoms. The value of temperature is needed for calculating heat capacity, liquid viscosity, and amorphous solubility. Molecular weight is used to calculate liquid viscosity and convert crystalline/amorphous solubility in log units to mass units (μg/mL). The number of carbon atoms and heteroatoms is needed for predicting the log P value based on Mannhold et al.'s method (eq 13). Boiling Point is left blank by default, but can be predicted by the JR method. However, it should be specified if the user knows its value beforehand, to predict a more reliable critical temperature, which requires boiling point as an input parameter (eq 4). The radio buttons offer the option to select the general solubility equation or its modified form. Once all necessary information is specified, a single click of *Compute Physical Properties* generates an output of the predicted results from the program at the bottom of the window. *Reset to Default* will restore the program to its initial state. If required input parameters are missing for the requested prediction of physical properties, such as data parameters are not available for predicting liquid viscosity for nitrogen and sulfur groups, a "Not Computable" will be displayed in the printed out results.

In the automatic mode, the users can allow the program to detect the types/occurrence of the functional groups in a molecule. To do this, the users should click "*Auto Detect Groups via SMILES*" in the main window. An interface as shown in Figure 3 will be displayed. The users now only need to type in the SMILES code that represents their molecule and specify temperature, boiling point, and the general solubility equation. After clicking *Compute Physical Properties*, the program will automatically parse the SMILES string, match the groups using SMARTS summarized in Table 2, remove duplicate groups, and determine types/numbers of groups before finally calculating the physical properties. If certain groups are not defined in the JR method, an error message will be shown. In addition, the users can write multiple SMILES strings into a text file and then load it by clicking "*Or Load Your SMILES File.*" Clicking on the "*Save Predicted Results*" button will direct the users to a dialog window that asks where to save the results as a spreadsheet (in .csv format). In the spreadsheet, if the parameters are missing in the JR database, the predicted results will be shown as "NA" in the cell. If some functional groups are not defined in the JR method, it will be shown as "Error" instead. If the program has trouble parsing the SMILES string, "NaN" will be stored into the cell.

ALGORITHM

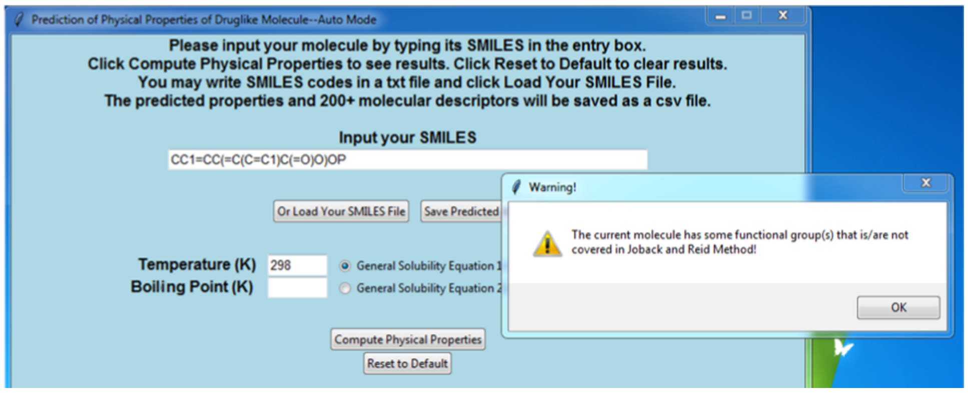
A schematic drawing that describes the operating procedures of JRgui program is displayed in Figure 4. The users have the options to use either a manual or an auto mode. In the manual mode, the program requires the input of functional group information by the user for the program to assign corresponding group contribution parameters from the JR database to predict the requested physical properties, which are provided at the bottom. If certain parameters are missing in the database, a "Not Computable" message will appear. In Python, this is implemented using a *try*: *except* block of code to handle errors and exceptions.[24] The program first executes the block of code in *try*; if no error occurs, then *except* is skipped, otherwise the *except* clause is executed.

In the automatic mode, after users provide a SMILES code, the program uses "search substructure" functionality from the RDKit package to parse the code based on the SMARTS of each 41 groups in the JR method. A summary of SMARTS codes for each group is listed in Table 2.



Groups (multiplicity)	Atoms (This work)	Atoms (Loosely defined SMARTS)
$-CH_3$ (1)	(0)	(0)
-OH(phenol) (1)	(10)	(10)
-COOH (acid) (1)	(7, 8, 9)	(7, 8, 9)
=CH< (ring) (3)	(2), (5), (6)	(2), (5), (6)
=C< (ring) (3)	(1), (3), (4)	(1), (3), (4)
Duplicate counts		
-OH (alcohol) (1)	-	(9)
>C=O (non-ring) (1)	-	(7, 8)
=O (1)	-	(8)
=C< (non-ring) (1)	-	(7)

Figure 5. Example on searching groups in 4-methylsalicylic acid. The numbers in the two columns in the right-hand side are atom positions of nonhydrogen atoms. Using SMARTS code as specific as possible, the JRgui program can find exact groups as expected. In contrast, the search with loosely defined SMARTS for functional groups returns duplicate counts as highlighted in red.


Prediction of Physical Properties of Druglike Molecule--Auto Mode

Please input your molecule by typing its SMILES in the entry box.
Click Compute Physical Properties to see results. Click Reset to Default to clear results.
You may write SMILES codes in a txt file and click Load Your SMILES File.
The predicted properties and 200+ molecular descriptors will be saved as a csv file.

Input your SMILES

CC1=CC(=C(C=C1)C(=O)O)OP

Temperature (K) 298
Boiling Point (K)

[✓ General Solubility Equation 1]
General Solubility Equation 2

Warning!

The current molecule has some functional group(s) that is/are not covered in Joback and Reid Method!

Figure 6. Warning message will be displayed if the SMILES string contains functional group that is not covered in JR method.

The functionality can also detect occurrence of each group and heavy atom positions in a molecule. To illustrate this, 4-methylsalicylic acid (with a SMILES code, CC1=CC(=C(C=C1)C(=O)O)O) is taken as an example. Its chemical structure is shown in Figure 5: it has a benzene ring (in this case, 3 =CH< and =C< ring groups each), one methyl, one hydroxyl (phenol), and one carboxylic acid group. Using specifically written SMARTS codes for Joback and Reid functional groups (Table 2), the JRgui program is able to find desired groups as expected. It is worth mentioning if using loosely defined SMARTS codes, such as [$(OH1)A] for −OH (alcohol), [CH0;A;X3;!R]=O for >C=O (nonring), [OX1] for =O, and [CH0;A;X3;!R] for =C< (nonring), the program detects −OH (alcohol), >C=O (nonring), =O, and =C< (nonring), which are unwanted duplicated groups, as highlighted in red in Figure 5. Although JRgui program uses SMARTS as specific as possible, to be on the safe side, the additional codes have been written to remove potential duplicate counts. The algorithm to remove duplicates is to first look at atom positions of heavy atoms, which are shown in the right column of the table. One can see that all of these extra functional groups are within the carboxylic acid group because their positions, (9), (7, 8), (8), and (7) are just a subgroup of −COOH (7, 8, 9). The next step is to remove groups if their atom positions are a subgroup of another group. Finally, the program records the remained groups and counts their occurrences before performing a calculation on properties.

The same algorithm can be extended to detect nonexistent JR groups in the molecule. In this case, when a SMILES code is given, the program automatically detects number/type of matched JR groups (with duplicated groups removed) and positions of all heavy atoms. If certain groups have not been incorporated in the JR method, such as a boron or phosphorus group, the sum of heavy atoms in all detected groups will differ from the total heavy atoms count in the molecule. If that happens, the program will display a warning message to alert the users. An example is shown in Figure 6, in which an addition of phosphorous to the SMILES code for 4-methylsalicylic acid brings up the warning message.

In the batch process for a series of molecules, the program uses a *for* loop to deal with each single SMILES code. A quick save to .csv format is made possible using the *pandas* module in Python, where it records data as a *DataFrame* and exports the data in various formats (.xls, .csv, .htm, .tex, etc.).[16b] The spreadsheet records 11 physical properties predicted by JR method and 4 solubility predictions for crystalline and amorphous compounds using general solubility equation and its modified form. In addition, as many as 187 molecular descriptors calculated by RDKit program are also summarized.

■ PERFORMANCE

The performance of JRgui program was tested on Karthikeyan's data sets, which contain 4450 SMILES.[25] On Windows 10 with Intel Core i7-4790 CPU, 3.6 GHz, and 16.0 GB memory, it takes ~9 min to finish all calculations (export 4450 × 201 = 894 450 data points to a spreadsheet); it takes ~12 min on a Windows 7 machine with Intel Core i5-5300U CPU, 2.3 GHz, and 8.00 GB memory, and on a macOSX 10.10.3 system with 3.1 GHz Intel Core i7 and 16 GB memory, the time taken is ~10 min. It is noted that failure to parse SMILES would slow down the process.

AVAILABILITY

JRgui program has been tested on various operating systems, including Windows 7, 8.1, 10, Ubuntu 14.04.5, and macOS 10.10.3, and is available for Windows, Linux, and macOS operating systems (all 64-bit systems). It can be easily installed using conda command in a terminal. More details about program installation and availability are listed in the home page of JRgui https://github.com/curieshicy/JRgui.

CONCLUSIONS

A new graphical user interface program, JRgui, which estimates the physical properties by classic group contribution method from Joback and Reid, is written using the modern object-oriented programing language, Python. The program operates in both manual and automatic modes. The detailed algorithms for building the program are explained. We hope our work could stimulate research in chemoinformatics with Python and its vibrant scientific community.

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acsomega.7b01464.

List of available descriptors in JRgui program (PDF)

AUTHOR INFORMATION

Corresponding Author

*E-mail: chenyang.shi@abbvie.com. Phone: +1 847-935-3328.

ORCID

Chenyang Shi: 0000-0002-2291-7325

Notes

C.S. and T.B.B. are employees of AbbVie and may own AbbVie stock. The design, study conduct, and financial support for this research were provided by AbbVie. AbbVie participated in the interpretation of data, review, and approval of the publication. The authors declare no competing financial interest.

ACKNOWLEDGMENTS

C.S. and T.B.B. thank various individuals from Drug Product Development (Research and Development) at AbbVie for their helpful insights and discussions. C.S. thanks RDKit and Conda community for helpful suggestions.

REFERENCES

(1) Fredenslund, A.; Jones, R. L.; Prausnitz, J. M. Group-contribution estimation of activity coefficients in non-ideal liquid mixtures. *AIChE J.* **1975**, *21*, 1086−1099.

(2) Nannoolal, Y.; Rarey, J.; Ramjugernath, J. Estimation of pure component properties Part 2. Estimation of critical property data by group contribution. *Fluid Phase Equilib.* **2007**, *252*, 1−27.

(3) Jankowski, M. D.; Henry, C. S.; Broadbelt, L. J.; Hatzimanikatis, V. Group Contribution Method for Thermodynamic Analysis of Complex Metabolic Networks. *Biophys. J.* **2008**, *95*, 1487−1499.

(4) Klincewicz, K. M.; Reid, R. C. Estimation of critical properties with group contribution methods. *AIChE J.* **1984**, *30*, 137−142.

(5) Joback, K. G.; Reid, R. C. Estimation of pure-component properties from group-contributions. *Chem. Eng. Commun.* **1987**, *57*, 233−243.

(6) Constantinou, L.; Gani, R. New group contribution method for estimating properties of pure compounds. *AIChE J.* **1994**, *40*, 1697−1710.

(7) Marrero, J.; Gani, R. Group-contribution based estimation of pure component properties. *Fluid Phase Equilib.* **2001**, *183−184*, 183−208.

(8) Leach, A. R.; Gillet, V. J. *An Introduction to Chemoinformatics*, 1st ed.; Springer Netherlands: Dordrecht, 2007; pp 1−255.

(9) Chemical Information Systems, Inc. http://www.daylight.com/.

(10) (a) http://ddbonline.ddbst.de/OnlinePropertyEstimation/OnlinePropertyEstimation.exe. (b) https://www.chemeo.com/. (c) http://www.molknow.com/Online/Estimation.htm. (d) Molecular Modeling Pro (Norgwyn Montgomery Software). http://www.norgwyn.com/mmpplus.html. (e) CRANIUM (Molecular Knowledge Systems). http://www.molknow.com/Cranium/cranium.htm. (f) http://umansysprop.seaes.manchester.ac.uk/tool/vapour_pressure.

(11) https://www.python.org/.

(12) Li, P.; Merz, K. M., Jr. MCPB.py: A Python based metal center parameter builder. *J. Chem. Inf. Model.* **2016**, *56*, 599−604.

(13) Zahariev, F.; Silva, N. D.; Gordon, M. S.; Windus, T. L.; Dick-Perez, M. ParFit: A Python-based object-oriented program for fitting molecular mechanics parameters to ab initio data. *J. Chem. Inf. Model.* **2017**, *57*, 391−396.

(14) Cao, D.-S.; Liang, Y.; Yan, J.; Tan, G.; Xu, Q.; Liu, S. PyDPI: Freely available Python package for chemoinformatics, bioinformatics and chemogenomics studies. *J. Chem. Inf. Model.* **2013**, *53*, 3086−3096.

(15) RDKit, Open Source Cheminformatics. http://www.rdkit.org.

(16) (a) https://docs.python.org/3/library/tkinter.html. (b) http://pandas.pydata.org/.

(17) Mannhold, R.; Poda, G. I.; Ostermann, C.; Tetko, I. V. Calculation of molecular lipophilicity: state-of-the-art and comparison of Log P methods on more than 96 000 compounds. *J. Pharm. Sci.* **2009**, *98*, 861−893.

(18) Yalkowsky, S. H.; Valvani, S. C. Solubility and partitioning I: Solubility of nonelectrolytes in water. *J. Pharm. Sci.* **1980**, *69*, 912−922.

(19) Jain, N.; Yalkowsky, S. H. Estimation of the aqueous solubility I: application to organic nonelectrolytes. *J. Pharm. Sci.* **2001**, *90*, 234−252.

(20) Hoffman, J. D. Thermodynamic driving force in nucleation and growth processes. *J. Chem. Phys.* **1958**, *29*, 1192−1193.

(21) Wildman, S. A.; Crippen, G. M. Prediction of physicochemical parameters by atomic contributions. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 868−873.

(22) Private communication with Jason Biggs.

(23) SMARTSviewer. http://smartsview.zbh.uni-hamburg.de/smartsview/view.

(24) https://docs.python.org/3/tutorial/errors.html.

(25) Karthikeyan, M.; Glen, R. C.; Bender, A. General melting point prediction based on a diverse compound data set and artificial neutral networks. *J. Chem. Inf. Model.* **2005**, *45*, 581−590.