##

Protein Structure Analysis and Design Laboratory
Michigan State University
http://www.kuhnlab.msu.edu
August 16, 2016
Joseph Bemister-Buffington and Leslie Kuhn

##

Binding Residue Alignment Tool (BRAT) version 1.0.0

BRAT is a tool that creates a sequence alignment highlighting equivalent binding site residues in two protein structures that are available as PDB files, based upon their structural alignment by Dali (http://ekhidna2.biocenter.helsinki.fi/dali/). The binding site in the query protein is defined by the user as either a list of residues or as all residues within a given distance of a ligand in the query PDB. In general, the "binding site" may be an interface with any other molecule, or it can be any set of residues the user would like to identify as structurally homologous in a second protein. Examples are a set of residues for which site-directed mutagenesis has been performed in the query protein, or the set of hydrophobic core residues when protein folding of homologs is being studied. BRAT is especially useful in cases where two proteins' sequences are not easily aligned (e.g., distantly related G protein-coupled receptors); by using their structural alignment as input, BRAT can structurally define the equivalent, functionally important residues. The Z-score reported by Dali should be used first to discern whether the two proteins are significantly similar; structures with significant similarities have a Z-score above 2.

This file has three sections:

(i) BRAT installation notes
(ii) Input pre-processing and how to run BRAT
(iii) Interpreting and using BRAT output

Section - I BRAT Installation notes

==========

BRAT runs using Python2.7.5 or greater (python3.5 is recommended)

To download and compile python, we recommend using Anaconda, freely available through Continuum Analytics at
https://www.continuum.io/downloads. Download the Anaconda graphical installer for Python3.5 and follow the instructions.

Section - II Input pre-processing and how to run BRAT

==========

Notes on preparing DALI alignment file for input to BRAT:

i) The first two lines are used to establish a label for the query and target in the alignment. BRAT will automatically take the last 4 characters of the top two lines as the labels. It is recommended that you use these lines to also identify which mol is what label. Thus all of the sample input contains "molecule 1: " and "molecule 2: " before the label as shown below.
ii) The next step is to align the two protein structures with the DALI pairwise alignment tool at:

http://ekhidna2.biocenter.helsinki.fi/dali/

(Please make sure to click on the "Pairwise" tab after following the link above.)

Example of the alignment file from sample 4:

```
molecule 1: 1htg
molecule 2: 1bai

# Structural equivalences
   1: mol1-A mol2-A     1 -   6 <=>    1 -   6   (PRO    1  - TRP    6  <=> LEU    1  - GLU    6 )
   1: mol1-A mol2-A     7 -  16 <=>    9 -  18   (GLN    7  - GLY   16  <=> ASP    9  - THR   18 )
   1: mol1-A mol2-A    17 -  35 <=>   29 -  47   (GLY   17  - GLU   35  <=> SER   29  - GLU   47 )
   1: mol1-A mol2-A    36 -  45 <=>   50 -  59   (MET   36  - LYS   45  <=> TRP   50  - ALA   59 )
   1: mol1-A mol2-A    46 -  59 <=>   63 -  76   (MET   46  - TYR   59  <=> GLN   63  - SER   76 )
   1: mol1-A mol2-A    60 -  67 <=>   78 -  85   (ASP   60  - CYS   67  <=> ASP   78  - ILE   85 )
   1: mol1-A mol2-A    68 -  99 <=>   92 - 123   (GLY   68  - PHE   99  <=> GLU   92  - ASN  123 )
   1: mol1-B mol2-B     1 -   6 <=>    1 -   6   (PRO    1  - TRP    6  <=> LEU    1  - GLU    6 )
   1: mol1-B mol2-B     7 -  16 <=>    9 -  18   (GLN    7  - GLY   16  <=> ASP    9  - THR   18 )
   1: mol1-B mol2-B    17 -  35 <=>   29 -  47   (GLY   17  - GLU   35  <=> SER   29  - GLU   47 )
   1: mol1-B mol2-B    36 -  44 <=>   50 -  58   (MET   36  - PRO   44  <=> TRP   50  - GLU   58 )
   1: mol1-B mol2-B    45 -  59 <=>   62 -  76   (LYS   45  - TYR   59  <=> PRO   62  - SER   76 )
   1: mol1-B mol2-B    60 -  67 <=>   78 -  85   (ASP   60  - CYS   67  <=> ASP   78  - ILE   85 )
   1: mol1-B mol2-B    68 -  99 <=>   92 - 123   (GLY   68  - PHE   99  <=> GLU   92  - ASN  123 )
```

The query pdb with known binding site should be submitted as mol1 and the homologous (target) pdb submitted as mol2. The PDB input files for DALI require a HEADER line to avoid skipping the first amino acid in the file. From the DALI results, copy the parseable data section for the highest scoring Z score alignment that includes the sequence of interest into a text file for use as BRAT input.

Notes on preparing binding residue input to BRAT:

The definition of the binding site (or other residues of interest) in the query protein, which will be mapped onto the homologous target protein by BRAT according to the DALI alignment, can be done in four ways, by providing:

i) A mini-PDB file containing only the ATOM lines of residues that comprise the binding site residues in the query protein, input with the -b command line option (shown under the section on running BRAT below). This mini-pdb file is intended to work with a standard pdb format (http://www.wwpdb.org/documentation/file-format), however, the mini-PDB need only have the

C-alpha entries for each binding residue.

```
TRPA117,THRA118,ASPA121,VALA122,VALA125,THRA126,PHEA201,THRA203,TYRA207,ALAA208
```

ii) Alternatively, the -b option will accept a list of binding site residues in a text or csv (comma-separated value) file containing only one line, with each binding residue ID separated by a comma, specified in the format:

<3 letter residue name><chainID><residue number>, <3 letter residue name><chainID><residue number>,...
Example of the binding residue input csv file from sample 1.

In the event that there is no chain for the query file, merely include a blank space in between the

```
TRP 117,THR 118,ASP 121,VAL 122,VAL 125,THR 126,PHE 201,THR 203,TYR 207,ALA 208
```

3 letter residue name and the residue number. The format will look like:

<3 letter residue name> <residue number>, <3 letter residue name> <residue number>,...
Example of a binding residue input csv file without a chain.

For the -b option, the file name should end in either .pdb, .txt, or .csv, according to the format for inputting the binding site residues.

iii) Alternatively, a ligand can be selected from the query file in order to determine binding residues. In order for BRAT to know which ligand is wanted, the user must input the ligand name, chain ID, and ligand number using the -l option. If the ligand information is input correctly, the binding site residues will be defined as all residues within a radius of any atom in the ligand. The default radius is 4.5 Angstroms but can be set to a different radius using the -r option; see -l and -r option below.
iv) If the ligand consists of multiple residues, a range of residue numbers plus chainID can be used to define the ligand, where the binding site residues will be defined as all those protein residues within a radius of any atom in this ligand. The default radius is 5 Angstroms but can be set to a different radius using the -r option; see -l and -r option below.

You can use any combination of one file format and one ligand specification. For instance, you can use the csv file and ligand span for input, but using the ligand span and ligand ID will not output the desired results.

Notes on preparing the query and target pdb files to input to BRAT:

i) BRAT requires the input file to be named with an extension ".pdb", e.g. 1ahb.pdb (not 1ahb.ent)
ii) The PDB files should be in standard pdb format in order to accurately take the residues within the desired radius of the user defined ligand. If only C-alpha entries or main chain atoms are present, it is likely that BRAT won't grab every residue within the desired radius.

The command to run BRAT for sample 4:

```
 /soft/linux64/python/2.7.5--GCC-4.4.7/python2.7
~/Binding_Residue_Alignent_Tool/brat.py
-q ~/Binding_Residue_Alignment_Tool/BRAT_sample/sample_4/1htg.pdb
-t ~/Binding_Residue_Alignment_Tool/BRAT_sample/sample_4/1bai.pdb
-a ~/Binding_Residue_Alignment_Tool/BRAT_sample/sample_4/1htg_v_1bai_DALI.txt
```

Must enter at least one of the following:
-b ~/Binding_Residue_Alignment_Tool/BRAT_sample/sample_4/1htg_binding_res.csv or 1htg_binding_res.pdb
-l G37,A,300 (if this ligand didn't have a chain in the pdb, input should look like G37,,300)
-s A,305-309 (if this ligand didn't have a chain in the pdb, input should look like ,305-309)

Running "/user/bin/python ~/Binding_Residue_Alignent_Tool/brat.py -h" will display the "help" menu with a list of valid options and a description of the input they require.

Section - III How to interpret BRAT output

==========

BRAT outputs the results of its alignment into an html file as well as a csv file of the binding residues annotated by how they were identified.

Note: In the file names below, <alignment_file> represents the prefix that precedes the ".txt" in the input alignment file.

(i) <alignment_file>_brat_aligned.html

This file contains the alignment based on the DALI alignment file. It can be opened with the internet browser of your choice. It identifies the chains for each alignment in one line above the actual alignment. The alignment shows in boldface the residues that were identified as the binding site residues in the target protein, based on structural equivalence to the user-defined query protein binding site residues. Hyphens show positions in which a protein has no structurally equivalent residues (gaps) relative to the other protein.

(ii) <alignment_file>_brat_res.csv

This file can be opened as text, or by Microsoft Excel, Pages, or LibreOffice Calc. This file is formatted to have two or four lines of residues depending on whether multiple binding residue identification methods were used. Each line identifies the method of identification for the residues then formats the residues to follow the input csv format.

Examples of BRAT runs with input and output files are present in the "/home/bemiste1/Desktop/python_scripts/Binding_Residue_Alignment_Tool/Input" and "/home/bemiste1/Desktop/python_scripts/Binding_Residue_Alignment_Tool/Output" directory for 6 alignments.

Thank you,

Joe Bemister-Buffington

Protein Structural Analysis and Design Lab
Department of Biochemistry & Molecular Biology
Michigan State University
E-mail: bemiste1@msu.edu