Manual for executing PharmRF scoring function

Prerequisites for running pharmrf.sh (script file):

1. fpocket program: please follow the instructions to install Fpocket locally in your computer.

http://fpocket.sourceforge.net/manual_fpocket2.pdf

https://sourceforge.net/projects/fpocket/files/fpocket-1.0/fpocket-src-1.0/

2. PLIP (Protein Ligand Interaction Profiler): please follow the instructions to install PLIP locally in your computer.

https://github.com/ssalentin/plip

3. WEKA: Download and install Weka v3.8 (stable version) according to your OS and system architecture.

https://www.cs.waikato.ac.nz/ml/weka/downloading.html

STAGE I: Steps to run pharmrf.sh

1. Create a folder in your Home directory and keep pharmrf.sh in the folder and enable Execute Permission (Properties -> Permissions -> Allow executing file as program).

2. Now create a sub-folder named "pdbs" and keep all the PDB files (protein-ligand complexes).

3. Generate a text file named "pdblist" with two columns: 1) PDB ID and 2) Ligand ID. The PDB IDs should be written in lower case only. An example of this pdblist is:

2r3q 5SC
4gcj X64
2r3r 6SC

Note the space between first and second columns. You can edit the pdblist provided in the examples available with this project to get the better hand.

4. Now open the terminal and type "bash pharmrf.sh" or "./pharmrf.sh"

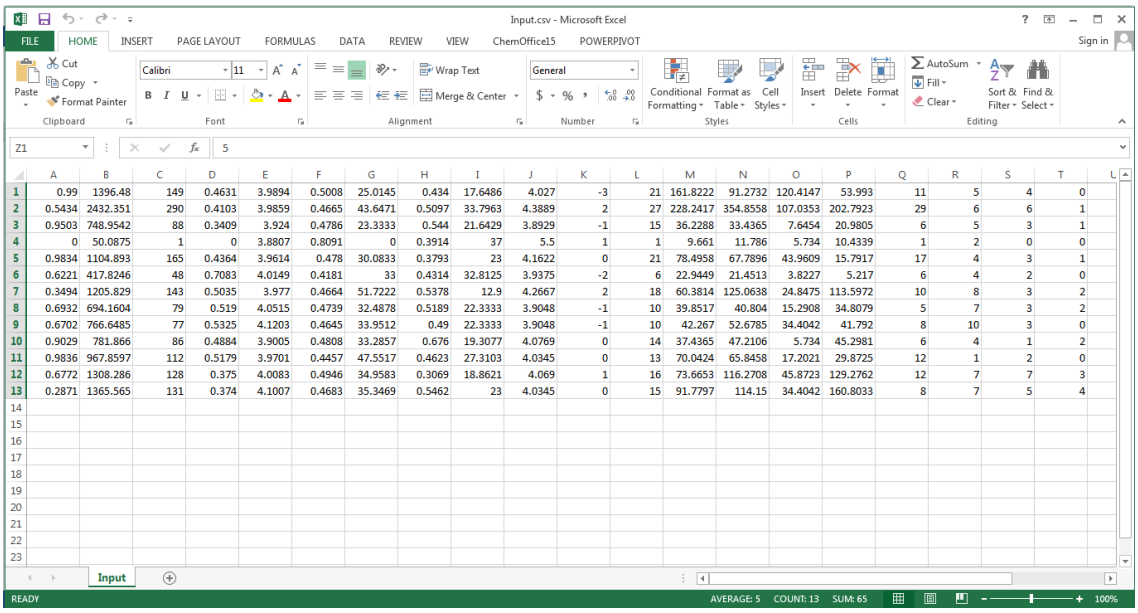
5. PharmRF will take one PDB entry along with the ligand ID code to locate the corresponding protein-ligand complexes in the "pdbs" folder at a time and initiate calculations. If both the Fpocket and PLIP successfully identifies the ligand from the ligand ID parsed through the "pdblist", it will indicate "1" in the terminal otherwise "0". It can be readily identifiable from the terminal which protein-ligand complexes were failed ("0"). A separate file "finalp.txt" which records these unsuccessful PDB entries is also generated.

Solution: You may need to look into the ligand code in the PDB file (structure file) and check whether there are alternative conformations for ligands are added or the chain ID is prefixed to the ligand code.

6. After the successful calculations, you can look at the results "results.txt" which is encoded in comma-separated text file (CSV) with the following form: PDB ID, ligand ID, 27 descriptor values. This is the input which you will feed to PharmRF WEKA model file to calculate the binding affinity.

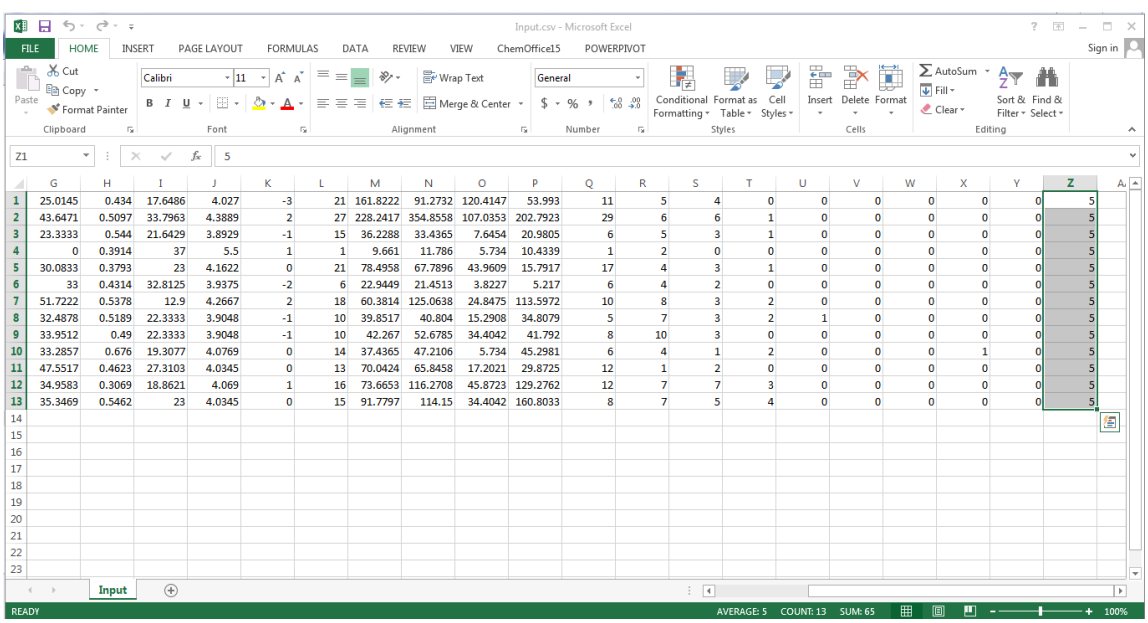
STAGE II: Steps to compile input file

7. Make sure to remove the two columns “as_density” (9th column in the results.txt, you can see the term “–nan” in this column) and “interChain” (6th column in the results.txt, you can notice all zeros in this column) from the “results.txt” using a Spreadsheet or MS Excel program as these two columns were not trained in the PharmRF WEKA model. The input now comes down to 25 columns. Have a look in the input file provided with the example set. Remove also the PDB ID and ligand ID from the “results.txt”. It will look like this below image. We will call this file as input_raw.csv



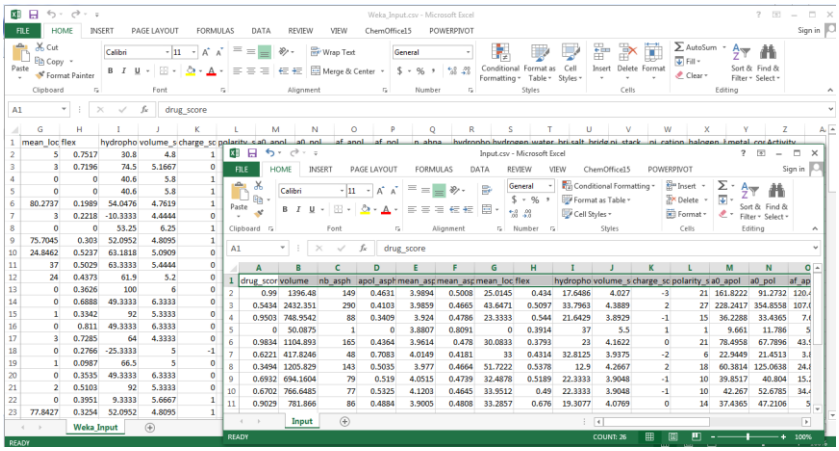
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	0.99	1396.48	149	0.4631	3.9894	0.5008	25.0145	0.434	17.6486	4.027	-3	21	161.8222	91.2732	120.4147	53.993	11	5	4	0
2	0.5434	2432.351	290	0.4103	3.9859	0.4665	43.6471	0.5097	33.7963	4.3889	2	27	228.2417	354.8558	107.0353	202.7923	29	6	6	1
3	0.9503	748.9542	88	0.3409	3.924	0.4786	23.3333	0.544	21.6429	3.8929	-1	15	36.2288	33.4365	7.6454	20.9805	6	5	3	1
4	0	50.0875	1	0	3.8807	0.8091	0	0.3914	37	5.5	1	1	9.661	11.786	5.734	10.4339	1	2	0	0
5	0.9834	1104.893	165	0.4364	3.9614	0.478	30.0833	0.3793	23	4.1622	0	21	78.4958	67.7896	43.9609	15.7917	17	4	3	1
6	0.6221	417.8246	48	0.7083	4.0149	0.4181	33	0.4314	32.8125	3.9375	-2	6	22.9449	21.4513	3.8227	5.217	6	4	2	0
7	0.3494	1205.829	143	0.5035	3.977	0.4664	51.7222	0.5378	12.9	4.2667	2	18	60.3814	125.0638	24.8475	113.5972	10	8	3	2
8	0.6932	694.1604	79	0.519	4.0515	0.4739	32.4878	0.5189	22.3333	3.9048	-1	10	39.8517	40.804	15.2908	34.8079	5	7	3	2
9	0.6702	766.6485	77	0.5325	4.1203	0.4645	33.9512	0.49	22.3333	3.9048	-1	10	42.267	52.6785	34.4042	41.792	8	10	3	0
10	0.9029	781.866	86	0.4884	3.9005	0.4808	33.2857	0.676	19.3077	4.0769	0	14	37.4365	47.2106	5.734	45.2981	6	4	1	2
11	0.9836	967.8597	112	0.5179	3.9701	0.4457	47.5517	0.4623	27.3103	4.0345	0	13	70.0424	65.8458	17.2021	29.8725	12	1	2	0
12	0.6772	1308.286	128	0.375	4.0083	0.4946	34.9583	0.3069	18.8621	4.069	1	16	73.6653	116.2708	45.8723	129.2762	12	7	7	3
13	0.2871	1365.565	131	0.374	4.1007	0.4683	35.3469	0.5462	23	4.0345	0	15	91.7797	114.15	34.4042	160.8033	8	7	5	4

8. Add a dummy column at the last of the input_raw file. This column is Activity and we are adding this dummy values so as to adhere to the input formatting of WEKA model. Finally, the input is complete by 26 columns (25 descriptors + 1 (Dummy) activity values). You can assign any activity values (e.g. 5).



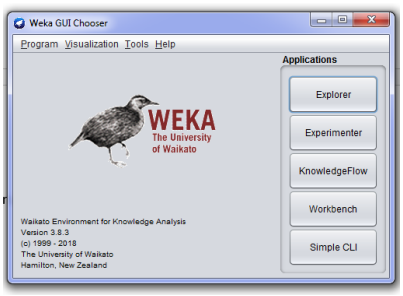
	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	25.0145	0.434	17.6486	4.027	-3	21	161.8222	91.2732	120.4147	53.993	11	5	4	0	0	0	0	0	0	5
2	43.6471	0.5097	33.7963	4.3889	2	27	228.2417	354.8558	107.0353	202.7923	29	6	6	1	0	0	0	0	0	5
3	23.3333	0.544	21.6429	3.8929	-1	15	36.2288	33.4365	7.6454	20.9805	6	5	3	1	0	0	0	0	0	5
4	0	0.3914	37	5.5	1	1	9.661	11.786	5.734	10.4339	1	2	0	0	0	0	0	0	0	5
5	30.0833	0.3793	23	4.1622	0	21	78.4958	67.7896	43.9609	15.7917	17	4	3	1	0	0	0	0	0	5
6	33	0.4314	32.8125	3.9375	-2	6	22.9449	21.4513	3.8227	5.217	6	4	2	0	0	0	0	0	0	5
7	51.7222	0.5378	12.9	4.2667	2	18	60.3814	125.0638	24.8475	113.5972	10	8	3	2	0	0	0	0	0	5
8	32.4878	0.5189	22.3333	3.9048	-1	10	39.8517	40.804	15.2908	34.8079	5	7	3	2	1	0	0	0	0	5
9	33.9512	0.49	22.3333	3.9048	-1	10	42.267	52.6785	34.4042	41.792	8	10	3	0	0	0	0	0	0	5
10	33.2857	0.676	19.3077	4.0769	0	14	37.4365	47.2106	5.734	45.2981	6	4	1	2	0	0	0	1	0	5
11	47.5517	0.4623	27.3103	4.0345	0	13	70.0424	65.8458	17.2021	29.8725	12	1	2	0	0	0	0	0	0	5
12	34.9583	0.3069	18.8621	4.069	1	16	73.6653	116.2708	45.8723	129.2762	12	7	7	3	0	0	0	0	0	5
13	35.3469	0.5462	23	4.0345	0	15	91.7797	114.15	34.4042	160.8033	8	7	5	4	0	0	0	0	0	5

9. We now have to add the column names to the input file so as to be recognized by PharmRF WEKA model. A CSV file, Weka_Input.csv is provided for this purpose. Just copy the first row which corresponds to column names and add to the top row of the input_raw file and save it as “Input.csv”.

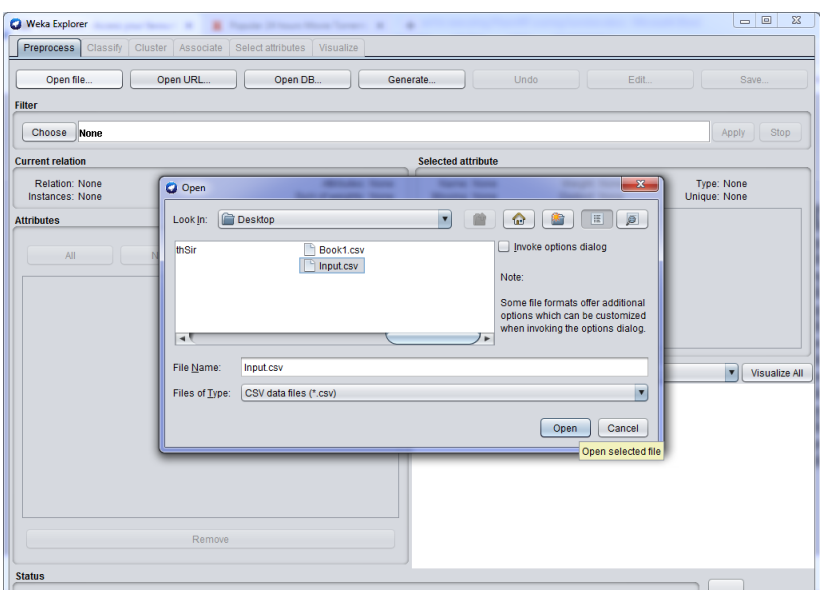


STAGE III: Steps to format input file for WEKA-compliance and compute binding affinity

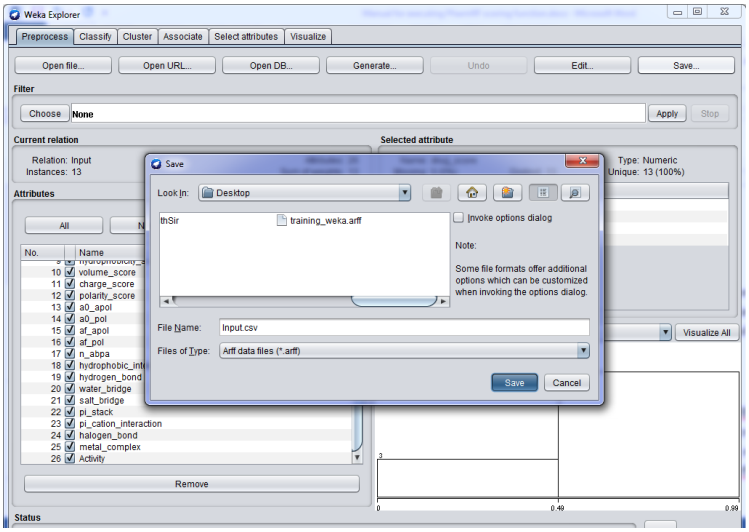
10. Open WEKA Gui Chooser and Click “Explorer” under Applications.



11. Now, load the CSV input file (input.csv) by Open file -> Input.csv. Remember to select CSV data files (*.csv) from the drop-down menu of ‘Files of Types’.

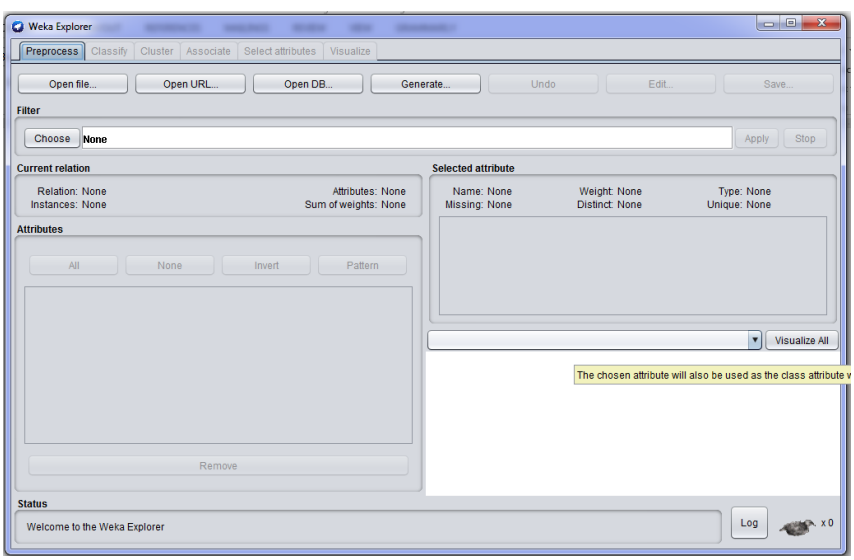


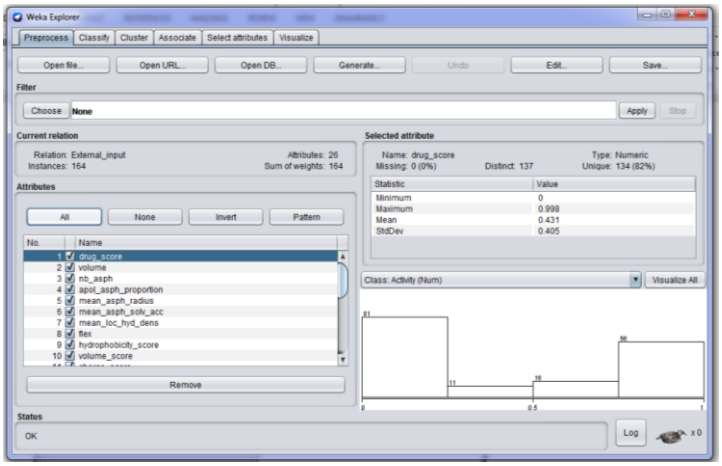
12. Now click “All” under Attributes option. We can see the 26 columns along with its column names have been successfully parsed by the Weka Preprocess module. This input file needs to be saved in ARFF format (Attribute-Relation File Format) readable by WEKA. Click “Save” at the top right panel and save it as Input.arff. We are safe to exit. Click “X” menu to close the Weka Explorer.



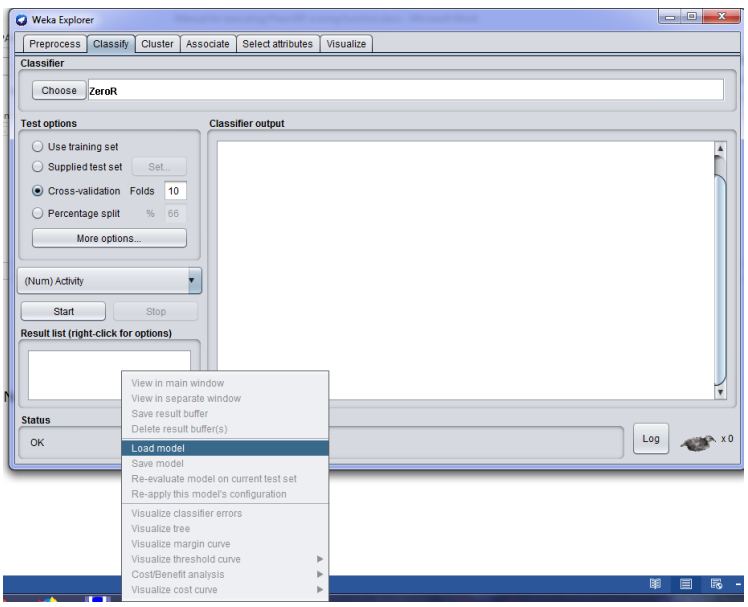
13. Again, click “Explorer” as we performed earlier.

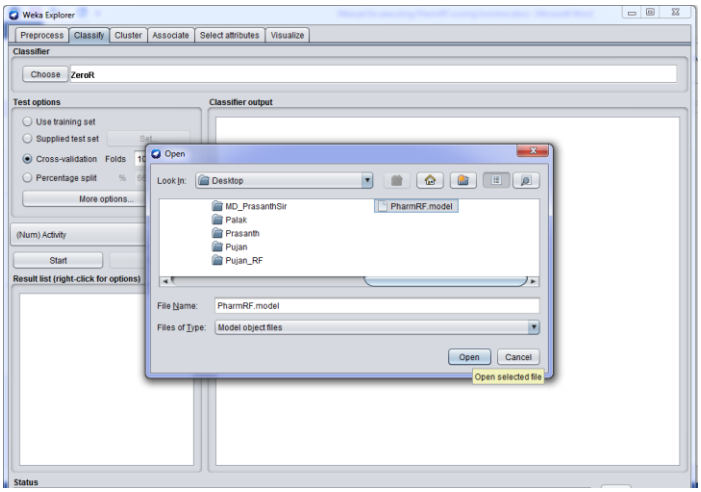
14. Open an external file provided with the example set so as to allow loading the PharmRF.model (model file) in the “Classify” tabs. This step is necessary to activate the “Classify” tab (first screen grab). Here, we will load the external input file which was used to externally validate the PharmRF model. For ease, we had provided the external file in ARFF format. Now, Open file -> load -> external.arff (second screen grab).



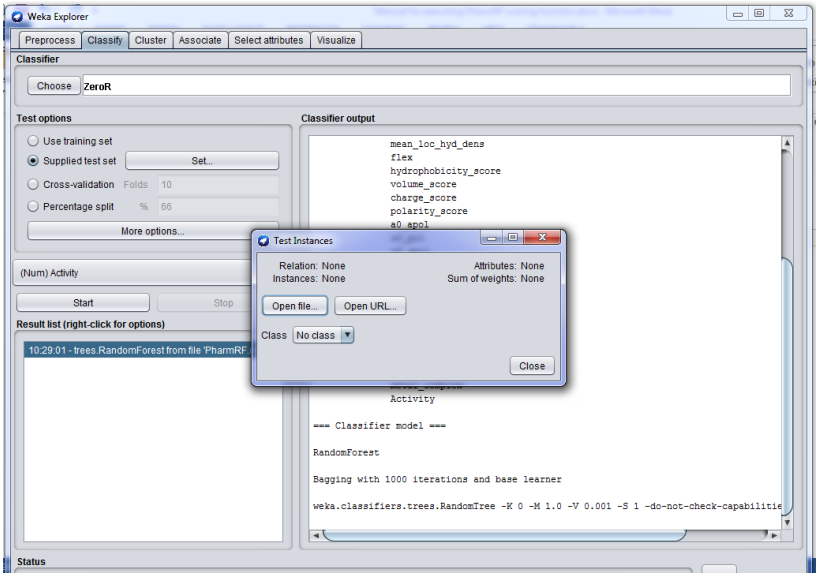


15. Now, the “Classify” tab is active for our calculations. Right click on the “Results list” white box and an option will be shown (first screen grab). Choose “Load model” and load the PharmRF.model file (second screen grab)

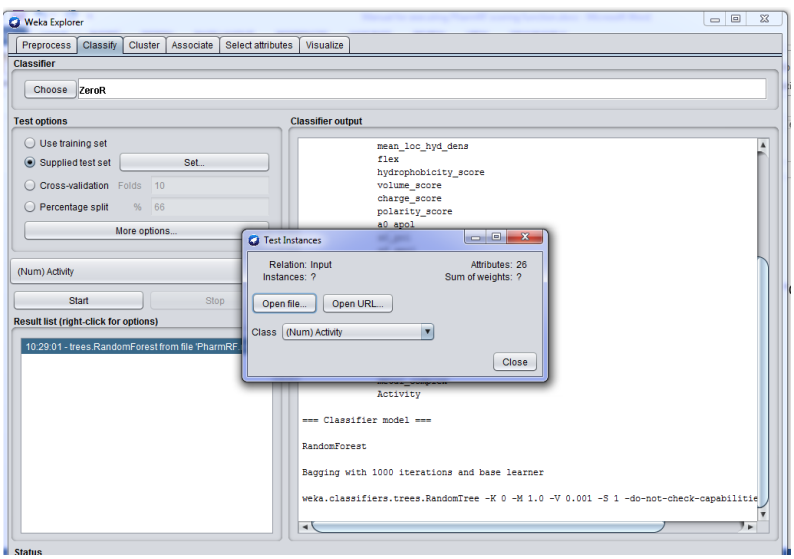




16. Now, select “Supplied test set” and click “Set”. It will be looking like this. We may now click “Open file” in the Text Instances dialog box and load the “Input.arff” file



17. After loading the input.arff file, the Text Instances dialog box will indicate the following terms: Attributes -26 and Class – (Num) Activity. “Close” this dialog box and now click “More options” under Test options panel in the Weka Explorer. A dialog box “Classifier evaluation options” will be opened.



18. In the dialog box “Classifier evaluation options”, perform the following changes step-by-step:

a. Go to output predictions, Click “choose” and select “PlainText”

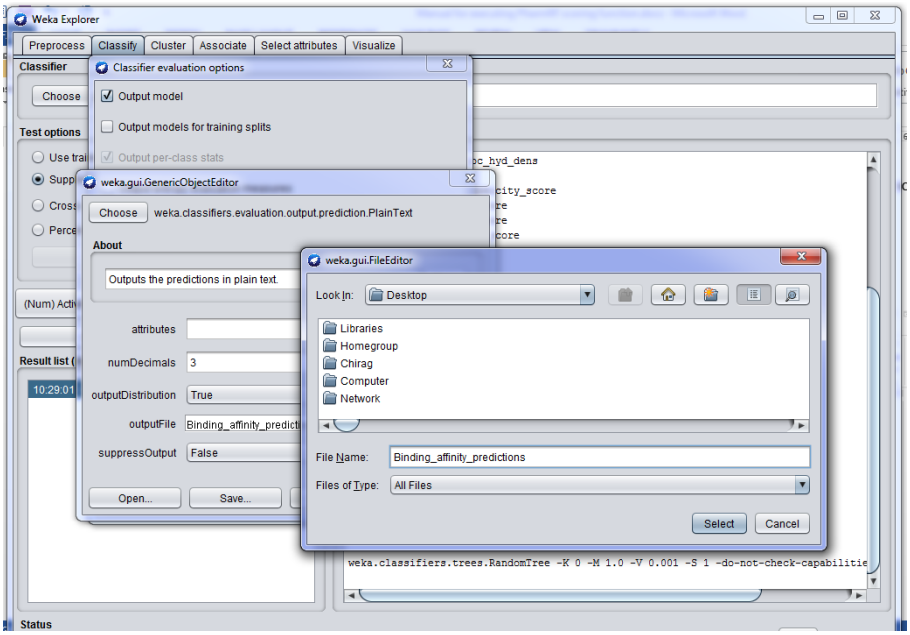
b. Just left-click in the white space adjacent to “PlainText” and weka.gui.GenericObjectEditor dialog box will be opened.

c. In this dialog box, select “True” for outputDistribution.

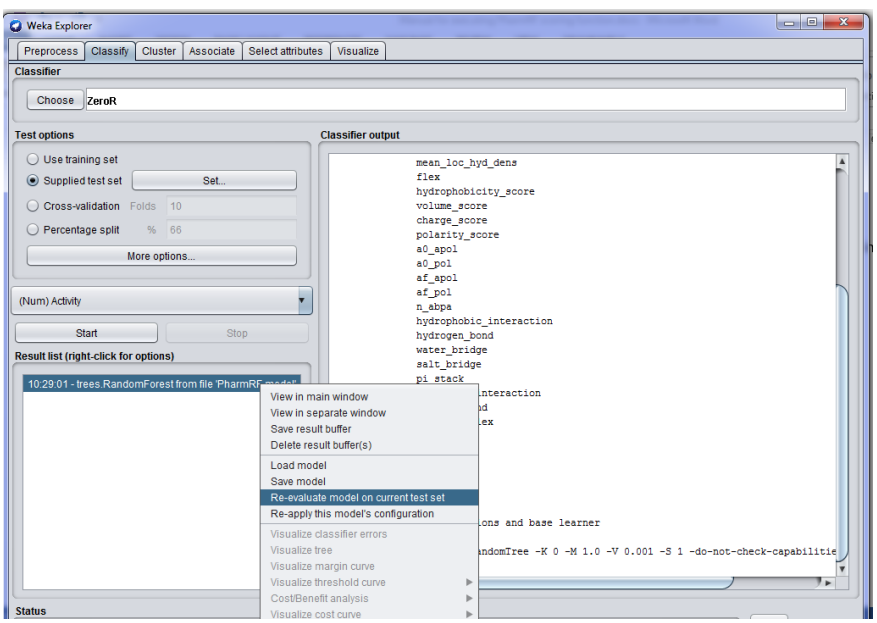
d. Click in the white space provided for outputFile which will open weka.gui.FileEditor box. See screen grab below.

e. Now, type “Binding_affinity_predictions” and click “Select”. This will close FileEditor box.

f. Click “OK” in the weka.gui.GenericObjectEditor dialog box as well as “OK” in the “Classifier evaluation options” to close these two remaining boxes.



19. Again, right click on the model “10:29:01–trees.RandomForest” from file “PharmRF.model” in the Results list and select “Re-evaluate model on current test set” option.



20. The results will be displayed in the white space of “Classifier output”. We can see the binding affinity predictions for the 13 protein-ligand complexes under the “predicted” column under “Predictions on user test set” (first screen grab). We should not worry about the “error” columns since we had provided dummy activity values for our input complexes. Similarly, correlation coefficient will also be 0 due to the bias (dummy activity values we provided). Note we can retrieve this results in the “Binding_affinity_predictions” PlainText file exported by WEKA (second screen grab). We can also note that inst# 13, 12, and 9 secured the top PharmRF scores which corresponds to PDB entries (“pdblist”): 1oit(HDT), 4fks(46K) and 4ek6(10K). 1oit(HDT) has an affinity IC_{50} value of 1-2 nM being potent inhibitor of CDK2. The protein-ligand complexes with the best PharmRF scores can be used for elucidating pharmacophores and perform pharmacophore-based virtual screening using any pharmacophore modelling softwares.