Implementation of Data Mechanics to Facilitate Pre and Post-Processing of Data for Protein-Protein Docking

Israel Desta

CS 504, Data Mechanics, Boston University, Boston, MA

Motivation

- Protein-protein docking is a computational tool, given the structures of the component proteins, predicts how they interact to form the complex
- ClusPro is one of the first web servers available for users to automatically search for the most feasible complex structure using thermodynamic and statistical mechanics principles
- ClusPro software actually assesses over a billion possible structures but gives access to the 70,000 most likely ones while the webserver only provides 10 models to users
- The 70000 possible structures are saved in an ftfile, and then clustered to give 10 or more clusters which are ranked by cluster size in a clustermat file where the center of each cluster is considered to be the representative model
- The ranking by size, though good, is not yet upto the golden standard, which is experimental data done by X-ray crystallography
- In order to improve the accuracy, it is important to:
    - Understand how ClusPro is currently performing on a well-accepted benchmark set
    - Find if there are special parameters that need to be used for different types of proteins
    - Understand the parameters that make a certain prediction more or less accurate
- Using the relational paradigm and Z3 linear SMT solver, I made the steps of pre-processing before docking the proteins, and the post-processing of the results more efficient
- To further highlight potential insights into parameters and how to improve the software, chart.js has been used to visualize different parts of the process

Workflow of Project

- Tools from the relational paradigm including selection, projection, aggregation, and intersection were used
- Introduction to provenance diagrams

Project 1
- Cluster Benchmark 5
- Rename PDB chains
- Find energy range, cluster population and homologs for a sample case

- Z3 SMT solver was used to solve a constraint satisfaction
- Correlation coefficients were calculated between fnat, iRMS, LRMS and the input DockQ score

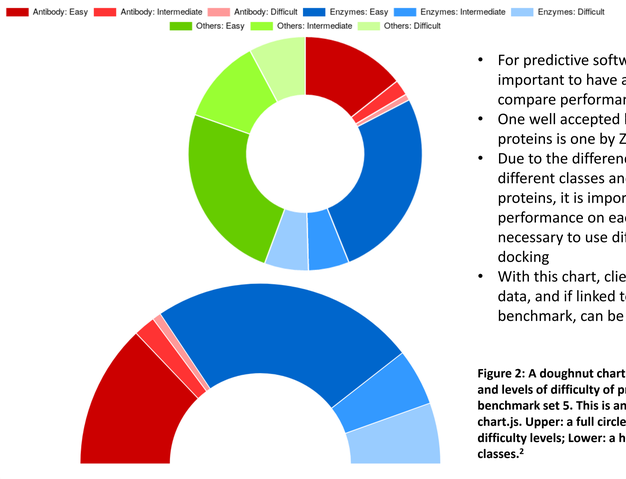
Project 2
- Solve for scaling factors of iRMSD and LRMSD to correctly evaluate predictions
- Statistical analysis of the evaluative parameters

- Chart.js was used to build interactive visuals for the input and obtained data
- A web-server was built using flask with visuals from chart.js and/or d3

Project 3
- Visualize the data worked with and obtained in Projects 1 and 2
- Make a web-server for client's access of interactive visuals

Figure 1: Outline of the three parts of the project with goals(right side) and tools learned from CS504 used (left side).

Input dataset



- For predictive softwares such as ClusPro, it is important to have a set of test cases to compare performances
- One well accepted benchmark set with 230 proteins is one by Zhiping Weng lab.[3]
- Due to the differences in nature between the different classes and difficulty levels of proteins, it is important to evaluate ClusPro's performance on each differently as it is necessary to use different parameters for docking
- With this chart, clients can see the spread of data, and if linked to the source link of the benchmark, can be very informative.

Figure 2: A doughnut chart showing the different classes and levels of difficulty of proteins in Zhiping's benchmark set 5. This is an interactive visual built using chart.js. Upper: a full circle showing all classes and difficulty levels; Lower: a half circle version with less classes.[2]

Project 1

1. K-means clustering benchmark 5 cases

- While the K-means clustering resulted in 10 true positives for difficult cases, and 97 true positives for easy targets, the clustering did not do so well with intermediate cases.

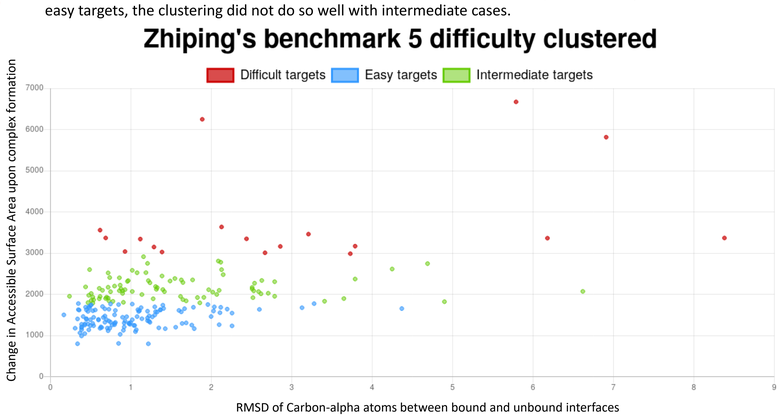


Figure 3: Interactive scatter plot built using Chart.js of K-means clustering done using relational tools.[2]

2. Renaming PDB files

- In order to check whether predictions are correct or not, predictions need to have the same chain names as the true structure.
- Since different experimental labs can obtains these structures, the chain names for the same objects can be different.
- Using relational paradigm, including projection, selection and more projection, I built a map of cases with their true complex structures and their chain names
- Then, I worked with one sample protein file, 1AHW, to change the chain names to prepare for docking

3. 1AHW: post docking look

- I counted the number of members in each cluster: [{'Center 605': 43}, {'Center 365': 20}, {'Center 24': 77}, {'Center 466': 26}, {'Center 555': 10}, {'Center 316': 75}, {'Center 770': 39}, {'Center 993': 13}, {'Center 942': 25}, {'Center 68': 11}, {'Center 251': 11}, {'Center 432': 21}, {'Center 969': 30}, {'Center 987': 3}, {'Center 678': 38}, {'Center 216': 28}, {'Center 434': 8}, {'Center 607': 38}, {'Center 789': 8}, {'Center 370': 27}, {'Center 453': 10}, {'Center 155': 21}, {'Center 576': 34}, {'Center 650': 31}, {'Center 898': 8}, {'Center 16': 70}, {'Center 828': 26}, {'Center 816': 15}, {'Center 694': 31}, {'Center 785': 5}, {'Center 264': 10}, {'Center 889': 71}, {'Center 983': 15}, {'Center 335': 28}]
- The ranges of the different types of potentials were also calculated from the ftfile (the main output of ClusPro) using relational tools for 1AHW predictions
- Finally, potential homologs to 1AHW were also pulled from the Protein Data Bank (PDB) using similarity features

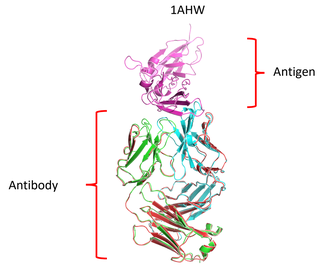


Figure 4: PyMOL rendering of protein 1AHW to show what the predictions of ClusPro look like

Project 2

1. Scaling factors using Z3

- The constraints used are

$$DockQ - 0.05 \leq \frac{fnat + \frac{1}{1+\left(\frac{iRMSD}{i}\right)^2} + \frac{1}{1+\left(\frac{LRMSD}{j}\right)^2}}{3} \leq DockQ + 0.05$$

- The constraints above are enforced for every single prediction of each protein (33 proteins in total) which means for about ~30 models for each protein which is solved 105 times comes to ~100,000 constraints
- The iRMS scaling factor was ~1.4 and LRMS scaling factor was 8.66 which are almost exactly as the true results reported by Basu et al.[4]
- The p-value scores showing the significance of correlation between the three evaluative parameters is shown below

Table 1: p-values from pairwise correlation coefficients of the four evaluative scores

	FNAT	iRMS	LRMS	DockQ
FNAT	0	0.023	0.106	0.004
iRMS	0.024	0	0.0185	0
LRMS	0.121	0.018	0	0.006
DockQ	0.005	0	0.011	0

Project 3

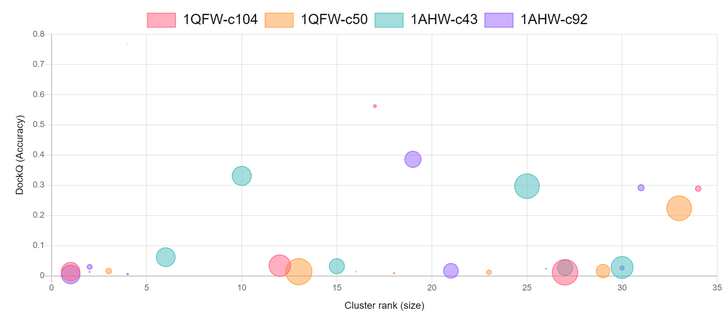


Figure 5: Combined data from project 1 and 2 to show the 3-D relationship between energy of predicted model (radius of circle) with relative cluster size (X-axis) and the accuracy/DockQ score (Y-axis).[2]

- For better visualization instead of absolute energy, the energy ranking divided by 50 was used as radius
- The X-axis also shows the relative cluster size (cluster rank) instead of absolute cluster size.
- Note that smaller bubbles should, as per ClusPro's hypothesis, be more accurate.

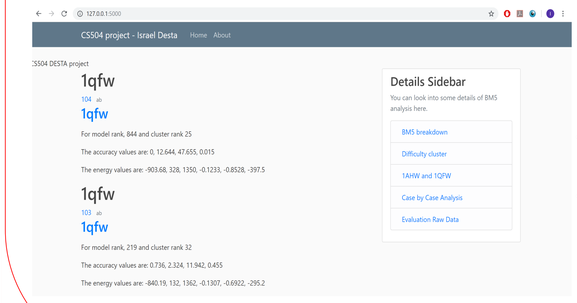


Figure 6: Screen shot of web server.[1]

- The graphics and visualizations shown from Figure 2, 3 and 5 will be uploaded on a web server shown on the side
- The website will allow clients to access the raw data as well as the graphical summaries
- Lastly, for figure 5, users will be able to choose to see protein by protein instead of groups of them

Conclusions and Future Works

- Using the relational paradigm I made the steps of pre-processing before docking the proteins, and the post-processing of the results more efficient
    - Clustered benchmark 5 protein cases based on change in ASA and change in C-alpha RMSD from bound to bound
    - Built a chain map to each protein case and renamed the component proteins of 1AHW
    - Analyzed the ClusPro outputs of 1AHW: found ranges of energy potentials from ftfile, counted the members of the clusters, and find homologs from Protein Data Bank
- Used Z3 SMT solver for solving a constraint problem and calculated correlation coefficients between the 4 evaluative scores to find which contributed the most to accuracy of prediction
- To further highlight potential insights into parameters and how to improve the software, chart.js has been used to visualize different parts of the process
- Finally, will be implementing a webserver for clients to view the interactive graphics using chart.js and d3. Besides containing figures 2, 3 and 5, it will allow users to choose specific proteins to visualize the details
- Potential steps forward is to scale the analysis to all proteins under study, i.e. instead of 1AHW only for counting cluster members, finding ranges and homologs, for all proteins in the BM5

Acknowledgments

Professor Andrei Lapets
Instructor Po-Yu Hsieh
TF Hao Chen

References

[1] Schafer, Corey. "Coreymschafer - Overview". *Github*, 2019, https://github.com/CoreyMSchafer. Accessed 29 Apr 2019.
[2] "Chart.Js | Open Source HTML5 Charts For Your Website". *Chartjs.Org*, 2019, https://www.chartjs.org/. Accessed 29 Apr 2019.
[3] Vreven, T. M., et al. J. Mol. Biol. **2015**, 427, 3031-3041
[4] Basu, S. and Wallner, B. PLoS One. **2016**, 11, e0161879