Evaluating the Efficiency of Rosetta Ab Initio and Numerical Linear Algebra in Protein Structure Prediction

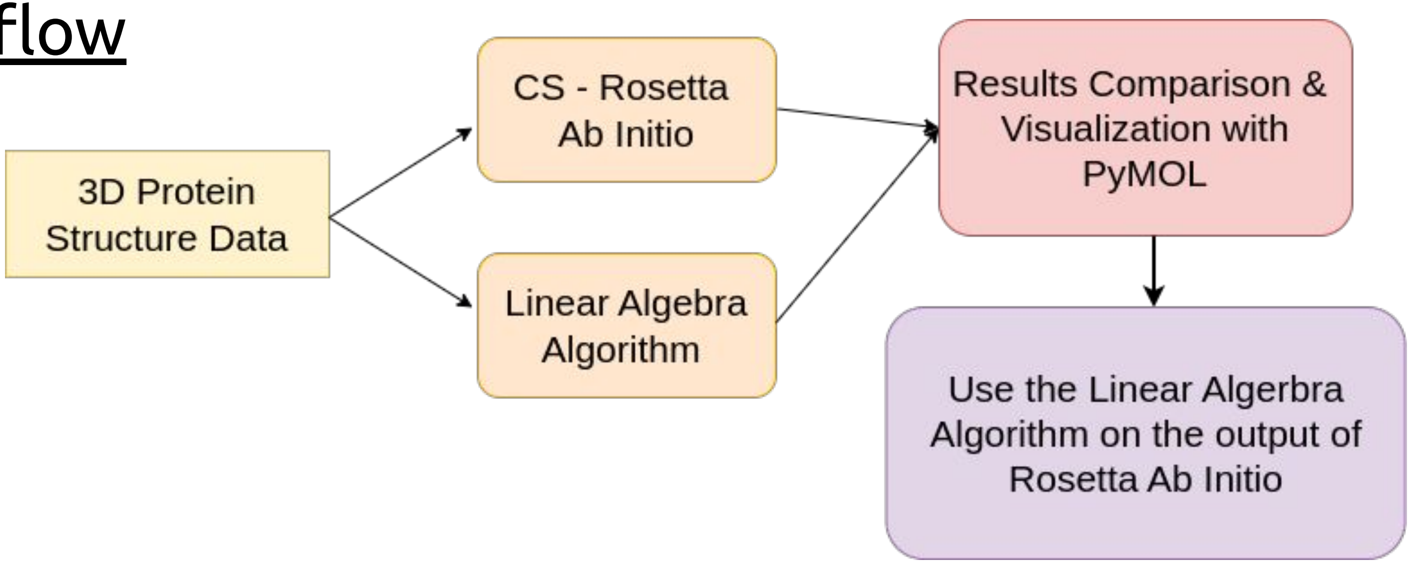
Spyros Alvanakis
Konstantinos Giatras

Course: Algorithms in Structural Bioinformatics

Aim

Assess and compare the performance of two protein 3D structure prediction software tools, Rosetta Ab Initio and Mr. Emiris' Linear Algebra algorithm, to evaluate their significance in determining protein structure, which is vital for drug design and disease understanding.

Workflow


3D Protein Structure Data
CS - Rosetta Ab Initio
Linear Algebra Algorithm
Results Comparison & Visualization with PyMOL
Use the Linear Algerbra Algorithm on the output of Rosetta Ab Initio

Rosetta Ab Initio

- **Input**: amino acid sequence (fasta file) plus 3 additional files acquired from the Robetta server
- **Monte Carlo algorithm**: making random, incremental modifications to the protein's structure and estimating the energy of each new conformation via a scoring function, aiming to minimize it
- **Output**: a silent file, containing all the pdb files of the produced predicted proteins of the run

Proteins used:

PDB Entry	Protein Name	Number of Amino Acids	Ensemble States
2JYV	Human Granulin F	32	10
2M0D	Miz-1 zinc finger 5	30	20
2MPC	Pyrin domain of human Pyrin	90	10
2MS8	Mitochondrial antiviral-signaling protein	102	20
6MWM	Bat coronavirus HKU4 SUD-C	81	20

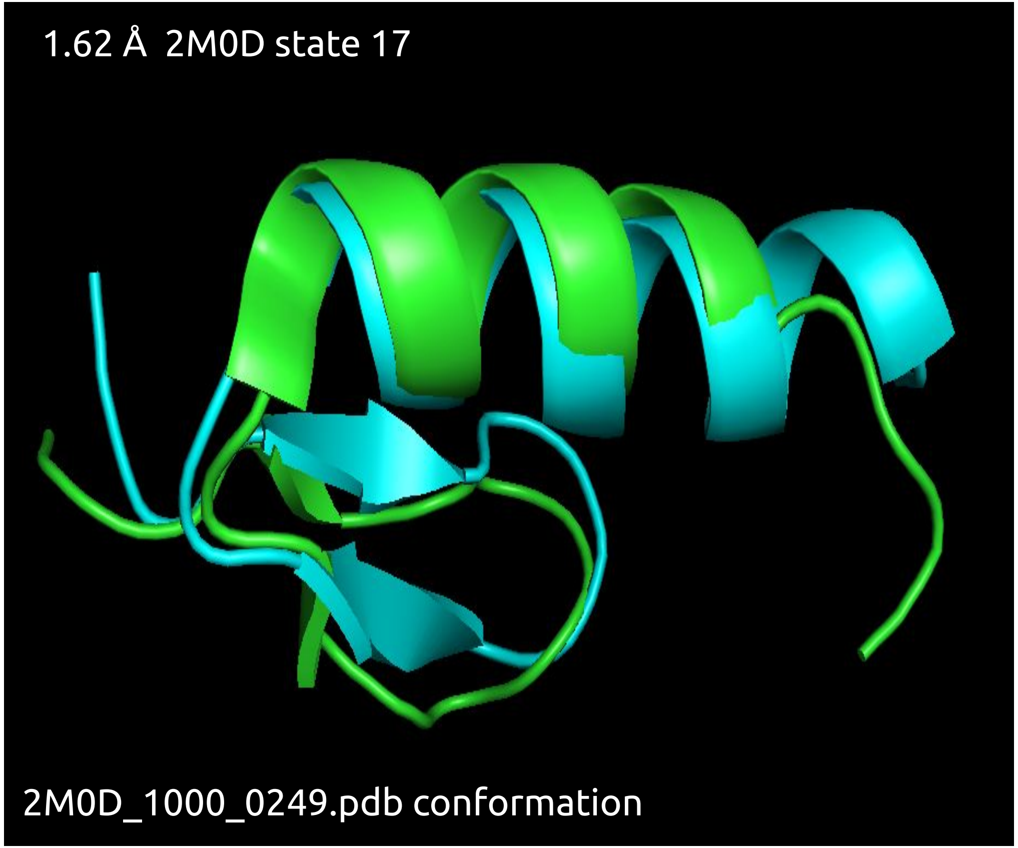
Rosetta Ab Initio

For each protein:

- Used to produce 10, 100 and 1000 predicted structures
- Terminal script that extracts the produced pdb files (from the silent file) and renames them for further more efficient handling
- Script that:
  - Calculates the Cα RMSDs between every predicted structure of the run and a state of the protein, and calculates the mean RMSD for this state
  - Repeats this for every state of the protein, and finds the minimum mean RMSD and the corresponding best protein state
  - Finds the predicted structure that has the minimum RMSD with the best protein state
  - Repeats the same process for every run (10, 100, 1000) of the protein

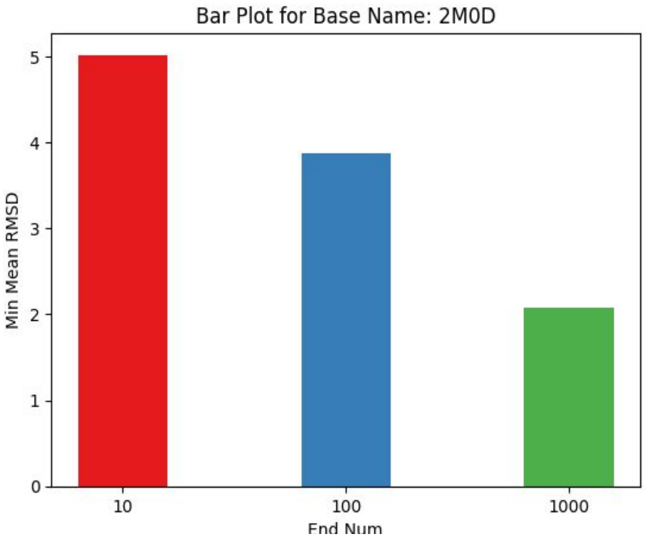
pdb_entry	run_num	min_mean_rmsd	best_state	min_rmsd_structure	min_rmsd
2JYV	10	5.6687963244	6	2JYV_10_0007	5.3762436692
2JYV	100	2.889796624	4	2JYV_100_0093	2.5849254506
2JYV	1000	3.067017334	4	2JYV_1000_0059	2.5655332268
2MPC	10	4.7445736298	2	2MPC_10_0008	4.498485885
2MPC	100	4.1856320657	3	2MPC_100_0089	3.992239566
2MPC	1000	2.6064312099	6	2MPC_1000_0053	2.4028245376
2MS8	10	7.4903670296	12	2MS8_10_0002	7.0814274566
2MS8	100	4.3998273458	18	2MS8_100_0099	4.0022640553
2MS8	1000	3.7742616655	7	2MS8_1000_0813	3.2615478098
2M0D	10	5.0173995735	4	2M0D_10_0004	4.0478528583
2M0D	100	3.8767165118	11	2M0D_100_0069	3.3050146592
2M0D	1000	2.0770254063	18	2M0D_1000_0249	1.6249936226
6MWM	10	11.1530588	19	6MWM_10_0008	11.0508184034
6MWM	100	7.2796141565	15	6MWM_100_0011	7.0625705772
6MWM	1000	5.7380785105	7	6MWM_1000_0174	5.4174648295

Example: 2M0D (1000 run)

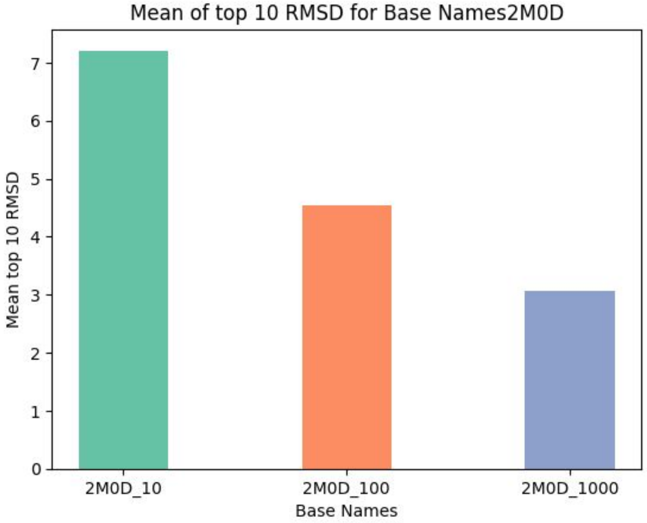


1.62 Å 2M0D state 17

2M0D_1000_0249.pdb conformation



Bar Plot for Base Name: 2M0D



Mean of top 10 RMSD for Base Names2M0D

Linear Algebra Algorithm

- **Input**: An NxN approximately symmetric or non-symmetric Cayley-Menger matrix (represents bounds or distances, originates from real or simulated NMR distance data) as a matlab file
- **Method**: Perturbs the matrix iteratively to reduce its rank, attempting to ensure that the sixth singular value falls below a given tolerance, thereby trying to produce a matrix with a rank of at most five (matrix embeddable in 3D space). The perturbations are guided by singular value decomposition (SVD) and implemented via a gradient-based approach to minimize the sixth singular value
- **Output**: An NxN symmetric Cayley-Menger matrix, that can be used to find the corresponding distance matrix and then extract the 3D coordinates of the backbone atoms

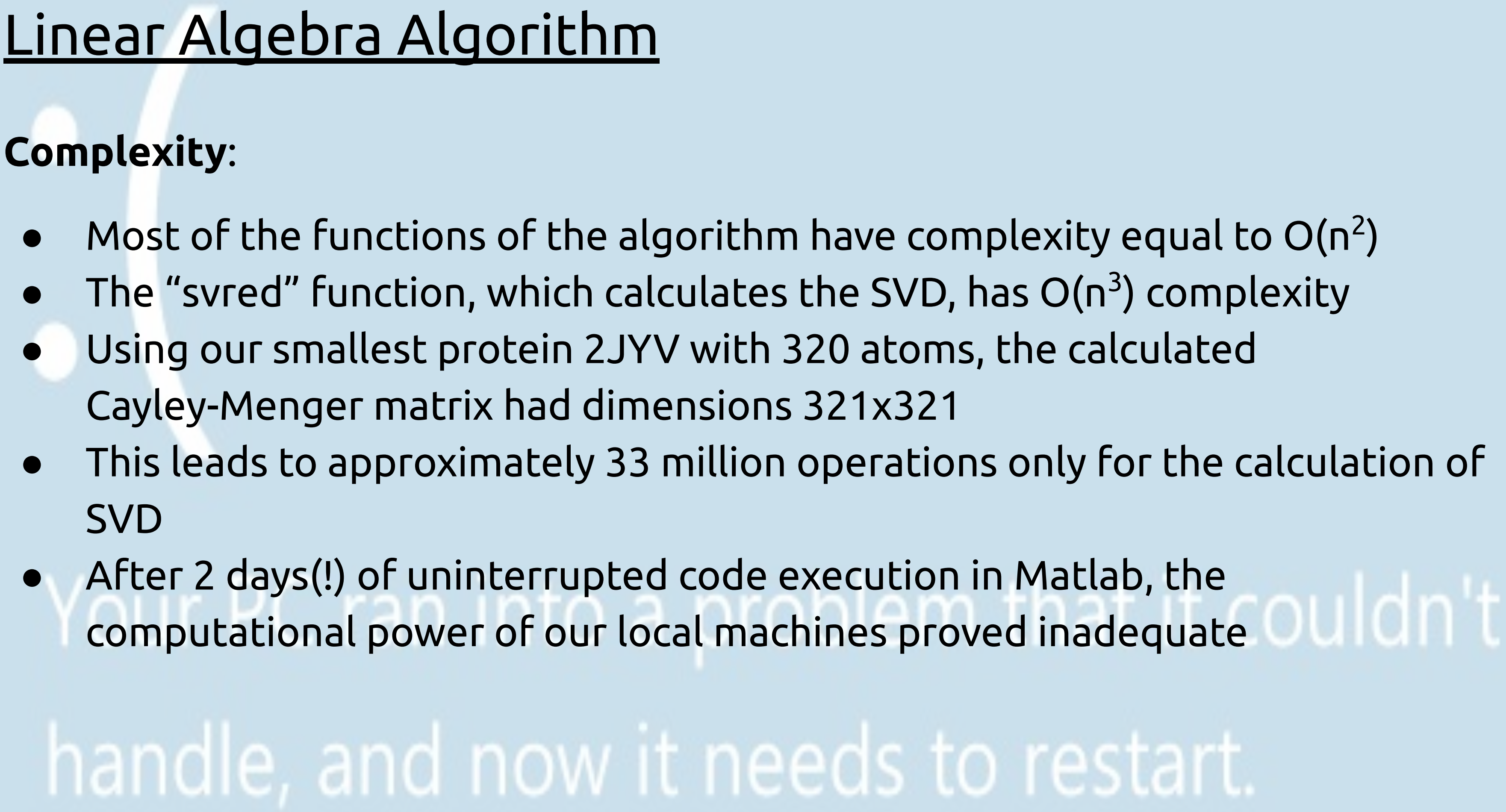
Linear Algebra Algorithm

- Performed a successful test run with the bounds7.m (7x7 matrix) file as input
- Created a script that:
  - Creates a distance matrix from the backbone atom coordinates of the best state of the ensemble of each protein
  - Creates a perturbation matrix from the distance matrix by adding noise of up to 2% to each distance value, generating intervals (simulated NMR data). The upper triangular portion contains the upper bounds, while the lower triangular portion contains the lower bounds of these intervals
  - Creates a Cayley-Menger matrix from the perturbation matrix and outputs it as a matlab file
- Attempted to run the algorithm in Matlab locally, with the 2JYV protein as input (321x321 matrix), but it was unsuccessful

Linear Algebra Algorithm

Complexity:

- Most of the functions of the algorithm have complexity equal to $O(n^2)$
- The “svred” function, which calculates the SVD, has $O(n^3)$ complexity
- Using our smallest protein 2JYV with 320 atoms, the calculated Cayley-Menger matrix had dimensions 321x321
- This leads to approximately 33 million operations only for the calculation of SVD
- After 2 days(!) of uninterrupted code execution in Matlab, the computational power of our local machines proved inadequate

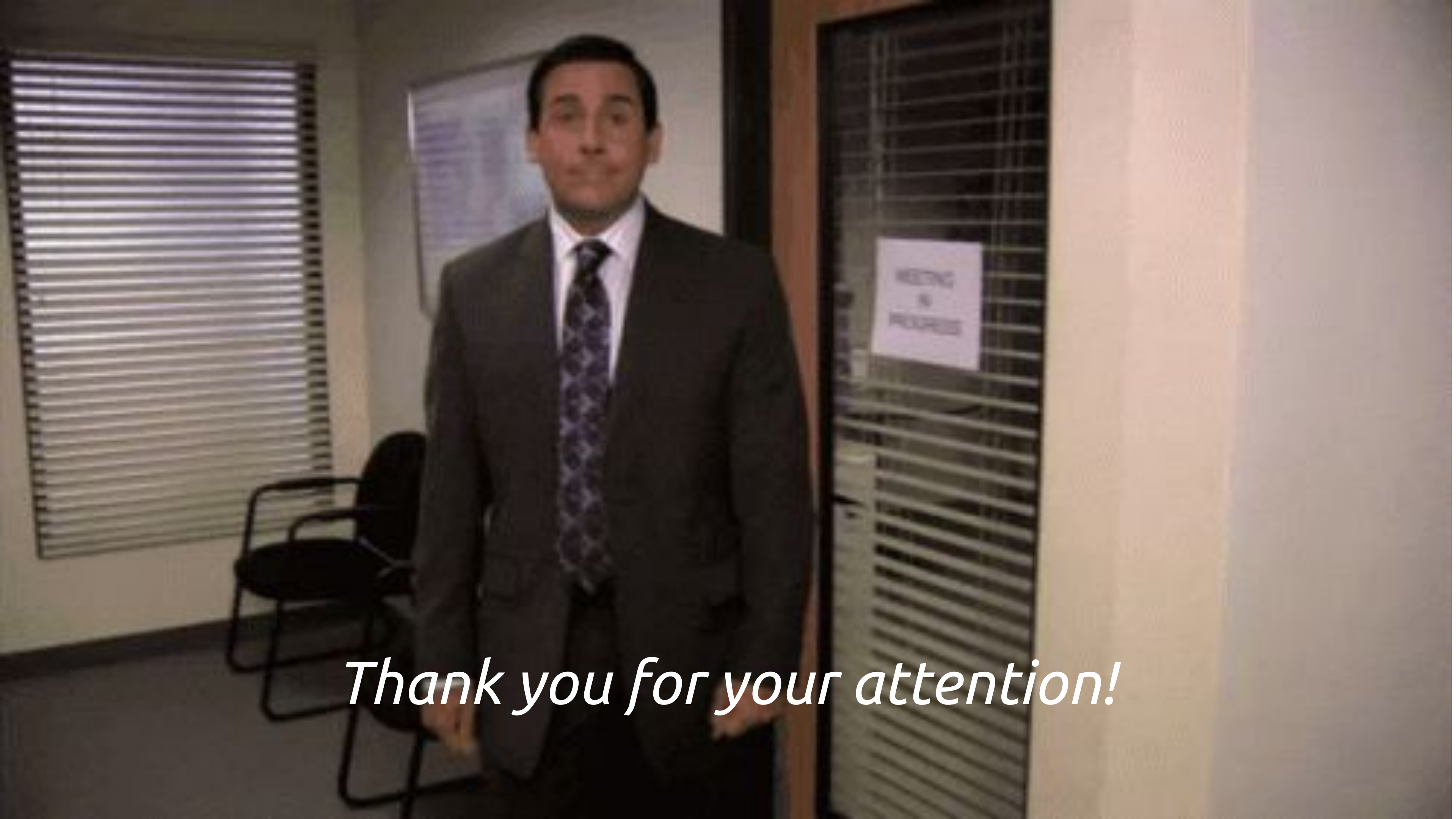


:(

Your PC ran into a problem that it couldn't handle, and now it needs to restart.

Conclusions

- From the Rosetta Ab Initio results, we observe significant RMSD differences depending on the number of produced predicted structures (up to 6 Å), concluding that the greater the number of produced predicted structures, the greater the accuracy of the prediction
- Limitations:
  - Rosetta Ab Initio algorithm cannot handle proteins with less than 30 amino acids
  - The Linear Algebra method seems to be unoptimized to handle larger proteins, due to its extreme runtime
- In conclusion, these two methods are not comparable and cannot be used it tandem for further improvement



Thank you for your attention!