Tea Freedman-Susskind

PI Mentor: Dr. Vanessa Jonsson

PhD Mentor: Natalie Dullerud

X July 2020

Second Interim Report

**Motivation**

Include these main points:

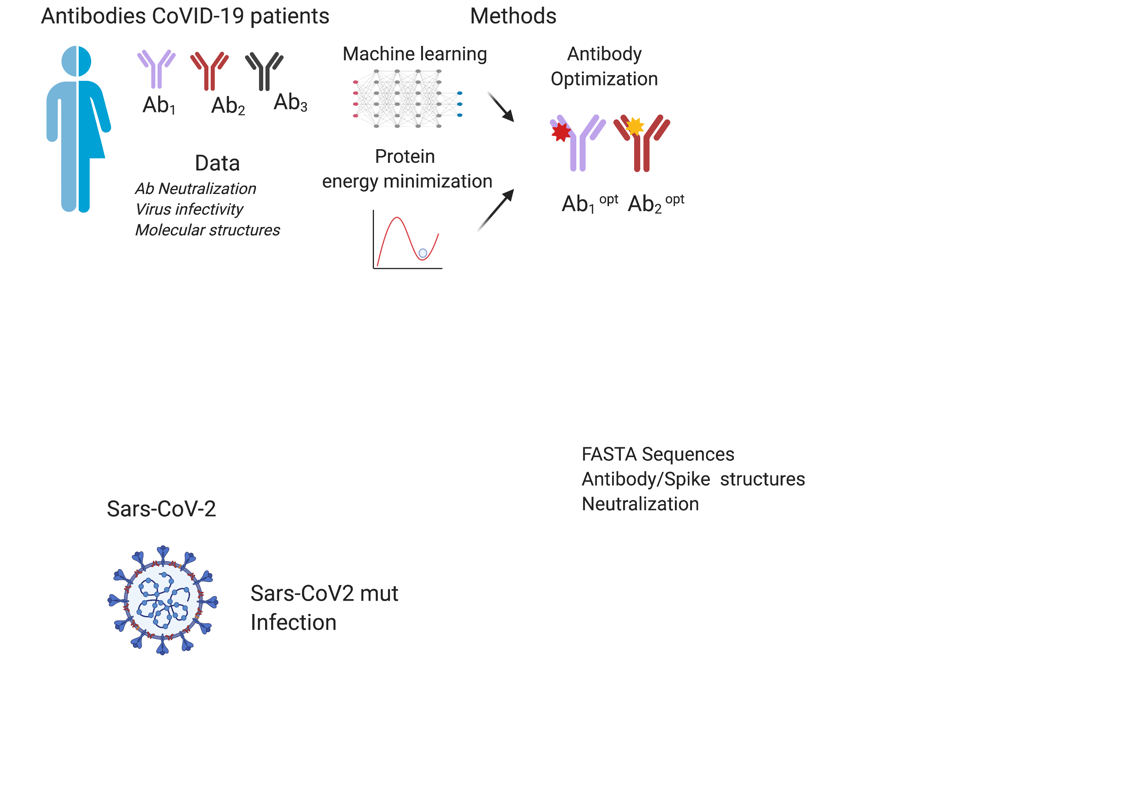
1. Big picture; we are in the process of developing a combined machine learning and computational chemistry analytic pipeline that leverages data from high throughput virus infectivity assays (bloom) and antibody neutralization studies (Robbiani, regeneron).
2. The goal is to predict antibody genomic locations that are likely to enhance neutralization of SARS-CoV-2 for the rational design of antibody optimization.
3. We do this with a combination of applying the new machine learning method developed by (dullerud and Jonsson) in combination with energy minimization methods on bound antibody-sars-cov2 molecular structures. (Fig 1).
4. We illustrate a preliminary version of this antibody pipeline with an example of optimizing the design of the anti-SARS-CoV-2 antibody C105 (PDB: 6XCM), whose structure was recently solved by (Barnes et al) in Pamela Bjorkman’s lab at Caltech.

Figure 1 | Overview of entire project. Machine learning algorithm developed by Jonsson lab

1. How we did this: we calculate differences in binding energies caused by mutations on the viral protein/cellular receptor and viral protein/antibody and estimate their impact on infectivity and neutralization. Specifically, we calculate the change in Gibbs Energy (ΔΔG) due to mutations and compare the difference between these ΔΔG values for the virus bound to the antibody and the virus bound to the human cell. Applying this analysis to the SARS-CoV-2 virus will yield useful information with potential application to analyzing neutralization potential of antibodies as well as to bolster the body of knowledge surrounding structural mutations of SARS-CoV-2.

**Progress**

**Results**

*Energy minimization calculations to optimize antibody design: an example with anti-SARS-CoV-2 antibody C105*

One or two sentences on the background of C105 discovery (Barnes paper).

To estimate how mutations of the virus might affect new antibody design, we calculated differences in Gibbs free energies between mutations of the virus and C105 antibody locations predicted by the satlasso algorithm. To do this we performed energy minimization on the structures and found several antibody genomic locations that could be amenable to optimization (Figure 2b). Talk about how you picked the top 10 locations (WT== true and ddg > 0 or WT == False and ddg <0). You focused on T28 as an antibody location to optimize and illustrate the pipeline. You generated new structures using build model to assess distance between C105 heavy chain and virus,.

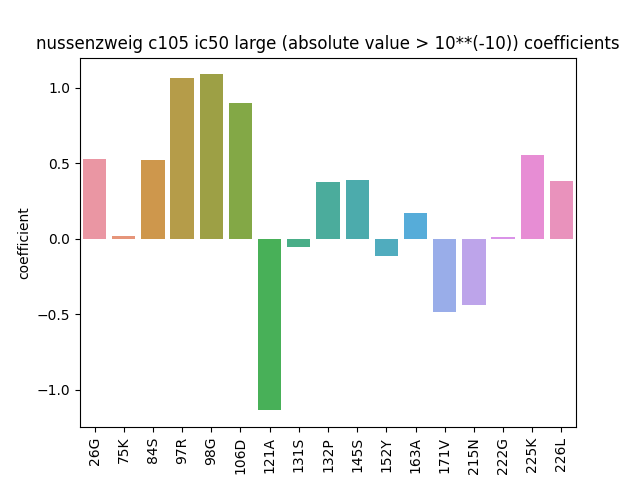


Figure 3 | a. Molecular structures with WT, and C105 heavy chain mutations T28I and T28D. Distances calculated using.

Figure 2 | a. Bar plot of coefficients relating impact of mutations to the C105 antibody to neutralization potency solved by satlasso regression (Dullerud and Jonsson). b. Box and strip plot of ddg energies calculated with FoldX for selected mutations of the heavy and light chains of C105 antibody defined in PDB structure 6XCM,

*Comparison of energy minimization calculations to experimental data for SARS-CoV-2 mutational analysis*

I tried to compare my calculated ΔΔG data with yeast-derived in vivo data to demonstrate its accuracy, but instead this just revealed underlying issues with how I was calculating ΔΔG—namely, that I was using a function that output ΔG instead of ΔΔG. I ran the function which produces ΔΔG data on old and new structures, all of which are SARS-CoV-2 spike protein fragments including its residue binding domain (RBD) bound to a variety of antibodies as well as the human ACE2 receptor. I used this analysis to identify aberrant mutations which either greatly enhance or detract from the bound complex’s stability. In addition, I furthered some analysis my co-mentor had undertaken of RBD mutations’ effects on the minimum concentration of that antibody necessary in vitro to neutralize half of spike proteins, the IC50 value, by plotting mutations of interest in a bar graph and analyzing them within the complex’s structure.

*Comparison of energy minimization calculations with evolved SARS-CoV-2 mutants in antibody selection experiments*

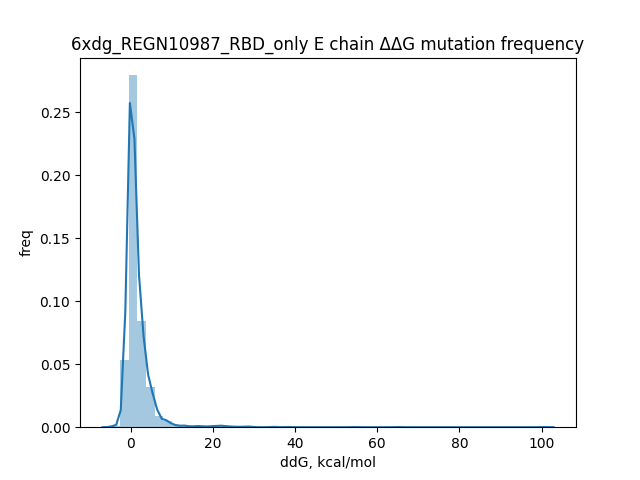
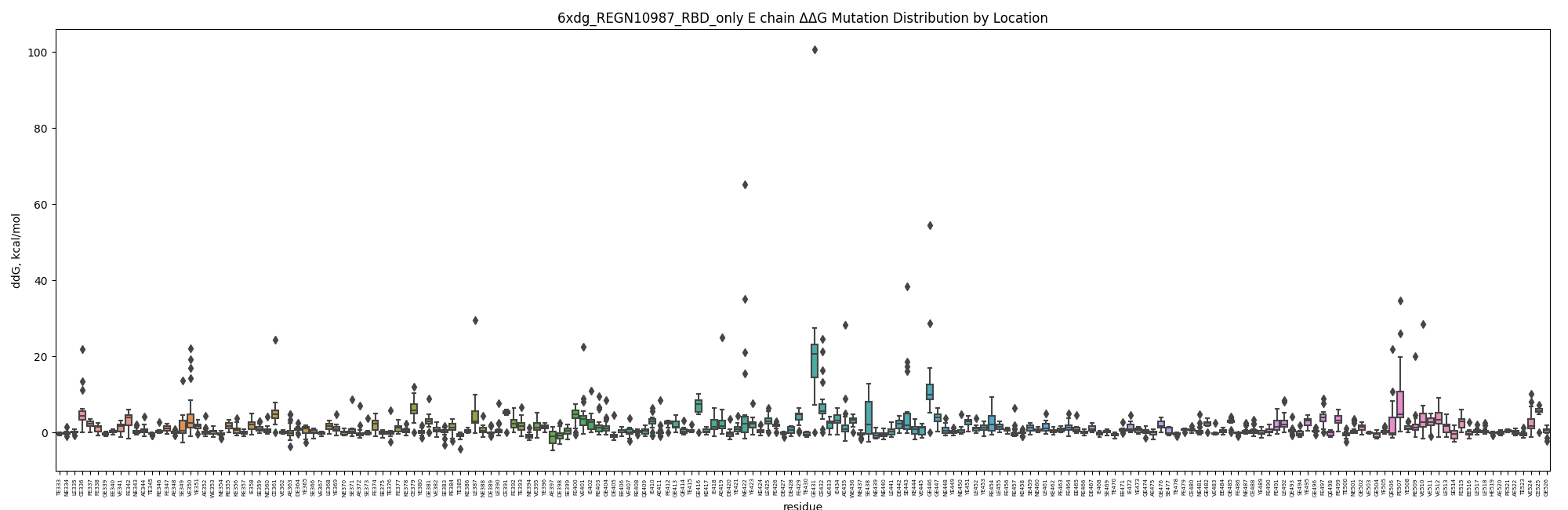
You need to say something about this if you are going to include it 

Figure 3|Box plot of Δ Δ G of mutations at each location of the RBD complexed with a REGN10987 antibody.

**Methods**

*Data availability and usage*

All PDBs used in this report. All neutralization and infection data used.

*FoldX calculations*

All the functions used in FoldX and parameters passed

*Molecular distance calculations*

*Software*

Describe the software you developed to do this.

I have continued to write scripts in Python to analyze data generated through FoldX, a Terminal-based program that simulates mutations and expected properties thereof using an energy minimization algorithm. In pairing simulated data with in vivo data from experiments in yeast, it became clear that there was a problem with my data, which proved to be the use of ΔG values from FoldX’s BuildModel function instead of ΔΔG values, which could be had from the PositionScan function (Starr et al). I proceeded to run all the structures I had previously collected ΔG data from through PositionScan to get ΔΔG, and reanalyzed the structures based on the now correct values. I used this opportunity to clean up my Python analysis scripts, and automate mutation running. In addition, while troubleshooting an especially slow mutation run, I discovered that PositionScan had been repairing the original protein file with each mutation, and so added in a repair command prior to running the PositionScan. I also added to the library of graphs generated using the Seaborn Python library for each protein file, creating a box plot and frequency plot for each chain in the protein structures and implemented automatic saving. I used some of these graphs to analyze data generated by my co-mentor on mutational improvements to and detractions of IC50 neutralization data for the C105 antibody, and explored its location in the structure using YASARA (cite 6xcm).

**Challenges**

The speed that mutation analysis runs at has proved a major limiting factor in my research. I alleviated part of this problem by repairing protein files prior to running the analysis, streamlining the process because a file repair job is run automatically with each mutation for an unrepaired file. I also set the FoldX task to the highest priority on my machine.

Another challenge has been fitting my calculated ΔΔG values with data from in vivo experiments. Investigating this problem led me to realize that I had been using the wrong values (ΔG) and the wrong FoldX function for my purposes. Additionally, I had been pairing data from the wrong protein file—the spike protein’s residue binding domain (RBD) complexed with a B38 antibody—instead of the file with a comparable complex, that of the RBD and the human ACE2 receptor.

**Future Work**

Before the end of the SURF program, I would like to finish running FoldX analysis to garner ΔΔG mutation data for all relevant complexes. I would like to pair this data with relevant in vivo data to demonstrate the technique’s effectiveness, and pick out and explore mutations of interests. I also hope to identify binding regions between antibodies and the spike protein’s RBD through analysis of distance between the molecules, and pair this distance analysis with ΔΔG analysis.