AS.410.712.81.SP13 Advanced Practical Computer Concepts for Bioinformatics

Steven Hwang

Final project

Spring 2013

I took a proteomics course last summer, and there were several tools available for analysis which required manual transfer of pertinent data by the user. I noticed tools, such as ProtParam <<http://web.expasy.org/protparam/>>, characterized proteins a single sequence at a time. My initial proposal hoped to make an example of a more integrative approach by providing several protein analysis tools in one location. I thought my initial proposal was a bit generic so I fine-tuned it a bit. I basically created a tool users can use for initial proteomic analysis of unknown amino acid sequences.

As you suggested, I parsed the remoteblast output and returned results as a JSON structure to the web interface. The user is then able to select the results that they are particularly interested in. The selected results are then converted to different odd codes before being stored in a MySQL database. The odd codes were presented in a way where the region of BLAST alignment between the query sequence and the reference sequence is in lowercase.

Odd codes can be particularly helpful in initial analysis of unknown sequence. For example, coiled-coil alpha helices (e.g. leucine zippers) are highly stable `supercoil` protein structures. A characteristic of these coiled-coil alpha helices are is a heptad repeat sequence motif (`HPPHCPC`) where `H` represents hydrophobic residues, `C` represents charged residues, and `P` represents polar/hydrophilic residues. In this case, the `functional` odd code generated by my project would be helpful in identifying any potential coiled-coil alpha helices within the query sequence or regions flanking the query sequence. If an area is identified, the user can also look at different properties by comparing several other odd codes displayed below and above the `functional` odd code.

There were several issues I had to overcome in order to complete this project. Some RemoteBlast hits were obsolete due to BLAST records being removed/altered. For example, the couple accession numbers below were hits via RemoteBlast which caused initial issues. I used the `get\_sequence` from Bio::Perl <http://doc.bioperl.org/releases/bioperl-1.0.1/Bio/Perl.html> function instead of `get\_Seq\_by\_acc` from Bio::DB::GenBank.

ZP\_15193172.1

http://www.ncbi.nlm.nih.gov/protein/ZP\_15193172.1?report=genpept

http://www.ncbi.nlm.nih.gov/protein/420712924?report=fasta

BLAST Results

ZP\_15322079.1

http://www.ncbi.nlm.nih.gov/protein/ZP\_15322079.1?report=genpept

http://www.ncbi.nlm.nih.gov/protein/420858313?report=fasta

Another issue I ran into was that the $seq object from Bio::SeqIO had an incomplete description ($seq->desc), which was a real pain when debugging the query scripts. It took me awhile to figure out what was going on, but this issue was resolved by simply parsing the saved text file of the selected RemoteBlast results.

In terms of results, I compared a couple RemoteBlast runs with the web BLAST interface results. The ranking of the hits were identical, confirmed by the order of the accession numbers, However, the max/total score and e-values differed, and I could not troubleshoot why because the query sequence, expected threshold value, and reference database were identical. I could not find any explanation via Google search, but I suspect the differences to be due to differences in the scoring matrix (BLOSUM62 vs BLOSUM45 vs etc.) used. I suspect the scoring parameters differed, but I did not have the time to confirm my theory.

Overall, this project was a great learning experience. I overestimated the task of this project and tried to accomplish too much (e.h. PHP was annoying to learn and bit awkward to use. I also felt documentation was sparse). Even though we covered a lot of material, I felt the manner in which the material was presented made it easy to absorb. Thank you again for another productive semester. I hope to keep in touch!