Project Ideas

Given below is a list of possible projects for you to work on. Some projects are better defined than others. But they are all interesting, and the only limitations are the amount of effort you put in and your creativity. If you wish to pick a project outside this list, please contact me as soon as possible. You should have picked something by Tuesday, February 3. Your choice has to be approved by me, since I have to make sure that there is no conflict with another group. Each group may have 1-2 members. In exceptional cases, I will allow larger groups. The class has both Biology and CS students. I strongly recommend finding a partner who knows more about the other field. Lot of the work is research-oriented and also result-oriented. I want to see some good results by the end of the semester. So start early. You are required to email me an update of your progress at least once every two weeks. Maintain a log file (or journal) containing your activities on this project containing: updates on your reading, progress on implementations and partial testing, ideas for future work, ideas that you may not be able to follow up, bug fixes, known current bugs in your code, organization of your program files and data files, etc.

At the end of the project, you will need to write a report (in doc or LaTeX format). It must include a short summary of your project. State clearly the following: title, group members, e-mail addresses, date, goals, hypotheses or assumptions, background with references and URLs, methods used (with references), what was implemented or achieved, summary of results, conclusions, possible future work.

Finally, prepare: (1) a 25-minute PowerPoint presentation of your work, (2) a handout to distribute to your classmates, (3) web page describing your project, and (4) a zip-compressed file containing your (commented) source code, data, results, report to be mailed to me. Your project should be completed and submitted by **April 7**. Your presentations will start from April 12.

Contact me for detailed information on the projects.

Transcription Factor Binding Sites

1. *Detecting Transcription Factor Binding Sites*: There are many tools available for detecting TF binding sites. A novel idea has been proposed to search for these binding sites in bacterial genomes. This involves using microarray data to perform more intelligent search for these motifs. This project involves implementing this idea and then testing it with known data.
2. *Transcription Factor Binding Sites*: There are many tools and databases available for detecting TF binding sites. A recent paper discusses their relative merits. This project involves applying several known tools on the available complete genomes. Compile your results and store them in our home-grown database for these binding sites. There are several different tools to choose from. And there are several different genomes to choose from. Whenever possible, confirm known binding sites and discuss new and interesting findings. (Collaborator: Chengyong Yang, Erliang Zeng & Prof. Kalai Mathee)
3. *Mining the database of Transcription Factor Binding Sites*: We have recently created a new database called PlasmoTFBM for storing lots of useful information related to transcription factor binding sites in the genome of the parasite *Plasmodium falciparum*. We want to know which of these motifs are also present in corresponding locations in the genomes of three other species of *Plasmodium* that have been sequenced. A whole-genome synteny map of four species of *Plasmodium* is now available (Shelby Bidwell’s work). Implement simple data

mining and visualization tools for the PlasmoTFBM database. (Collaborator: Chengyong Yang, Erliang Zeng & Prof. Kalai Mathee)

Sequence Analysis

4. *Analyzing unknown DNA fragment from unspecified bacterium*: Ms. Einstein isolated a mutant bacterium that was resistant to increased amount of antibiotic X. She narrowed down the search for the gene(s) responsible for this behavior and sequenced a DNA fragment. She has now sought your help. For her sequence, figure out: ORFs, ribosome-binding sites, promoter regions, termination regions, inverted repeats that may serve as regulatory binding sites, restriction sites, CpG islands, genes and the corresponding proteins, hydrophobicity plots (& interpretations), homologous genes and proteins, secondary structures in the protein, protein motifs, at least 10 possible functional annotations, restriction sites, protein structures using some homology modeler, discrepancies in your information, and any other new things you can think of. You may have up to 5 different sequences. (Collaborator: Prof. Kalai Mathee)
5. *Intron Analysis*: This project involves building visualization tools for the following problem. Given a multiple sequence alignment of a family of proteins from several different organisms, display the sequence alignment with information on the location of the introns in the corresponding genes. (Collaborator: Prof. Sawsan Khuri)
6. *Whole genome analysis*: Perform the whole genome analysis of Mycoplasma genitalium G-37. This is the smallest bacterial genome that has been sequenced. Analyze the genome for genes that are present in all bacterial genomes. Identify genes that are unique to this genome. Alternatively, pick the most recently sequenced bacterial genome and perform the same analysis.
7. *Whole genome comparison*: Given the entire genomic sequences of 2 related organisms, test and compare a variety of tools to compare them. What information can you extract and what tools can you use? More specific goals are:
   - Identify homologous genes. Given a gene, identify its homologue in the other genome. Compile list of genes with no homologues.
   - Study differences in the order of genes.
   - Study differences in functional annotations.
   - Compare and contrast available tools.
8. *Whole genome assembly software*: Given the entire genomic sequences of 2 related organisms, test and compare a variety of tools to compare them. What information can you extract and what tools can you use? More specific goals are:

Analysis using Gene Ontology Databases

9. *Gene Ontology problem*: Assume that somebody has done some genomic analysis and has listed a set of “genes of interest”. What can you say about these genes based on ontology information? Are some functions significantly enriched in this set? In other words, is there a significant representation of genes that are involved in, say DNA repair? Survey and test the tools available for this task. Tools include GOMiner, FuncAssociate, FunSpec, ProToGo, GoSurfer, goTermFinder, etc.

Detecting Recombination in sequences

10. *Recombination Detection*: Before performing phylogenetic analysis, it is important to know which sequences being analyzed are the result of

recombination, since in this case many phylogeny tools are not directly applicable. VisRD is a recent tool for detecting the presence of recombination in a set of sequences. However, it does not tell you which sequences are recombinants, which sequences they were recombined from, and the location of the breakpoints in the recombinant sequence. Modify VisRD to add these features. This has applications to serially-sampled within-host HIV sequence analysis. (Collaborator: Patricia Buendia)

Pattern Discovery

11. *Using negative examples*: Find a good data set with many positive and negative examples for the training set. Use machine learning tools to learn from such data.

Genetics Data processing

12. *Identify the alleles in a sample*: Use a learning program to learn from mobility data of known alleles and to identify alleles in unknown samples. (Collaborator: Prof. David Kuhn)

Projects related to Protein Structure

13. *Sequence-Structure Alignment*: Given two protein structures, design a sequence alignment algorithm that is meaningful. It should highlight (a) the positions in the alignment that are “aligned” in the sense of amino acid similarity, (b) the positions in the alignment that are “aligned” only in the structure alignment, and (c) the positions in the alignment that are not aligned structurally. (Collaborator: Tom Milledge)

I/O Optimization for Bioinformatics Tools

14. *Storage Optimization*: Many applications require the access to large databases containing biological information. Thus the efficiency of these applications depends on the efficient storage and retrieval of information. Several public domain bio-databases currently store hundreds of GB of biological data. The first step for this project would be to obtain the IO signature of a popular BioInformatics tool like BLAST and then suggest how IO performance can be improved. (Collaborator: Prof. Raju Rangaswami)

Microarray Topics

15. *Clustering*: Devise and test new machine learning tools (clustering or classification tools). (Collaborator: Prof. Tao Li)
16. *Fractal Analysis*: Can microrray data be modeled as a fractal? Implement and test an appropriate hypothesis. (Collaborator: Gaolin Zheng)

Genome-specific databases

17. *Create a DB*: Using the resources at http://www.gmod.org/, http://yourdb.org/, and http://www.gusdb.org/, create your own genome-specific DB for a simple organism (say *P. aeruginosa*). What are the features provided by the resources? What needs to be implemented? What can be added to it? Study the known examples available at:
    - ApiDB http://apidb.org
    - CryptoDB http://www.cryptodb.org
    - TcruziDB http://tcruzidb.org

- ToxoDB http://toxodb.org
- PlasmoDB - http://plasmodb.org
- YourDB - http://yourdb.org
- Toxoplasma gondii apicoplast genome http://www.sas.upenn.edu/~jkissing/toxomap.html

Miscellaneous Topics

18. *Design of Nested Primers*: Modify DePiCt to design nested primers for R-genes from *Arabidopsis thaliana.* (Collaborator: Prof. David Kuhn)
19. *Primer Design*: Design primers for specific forensic applications. (Collaborator: Prof. Bruce McCord)
20. *Probe Design*: Design good set of probes for specific forensic applications. (Collaborator: Prof. Bruce McCord)
21. *Improved Substitution Matrices*: An improved substitution matrix for BLAST searches is perhaps possible by using updated sequence data and by adding structural information. Implement this idea. (Collaborator: Prof. Sawsan Khuri)
22. *New Topics*: This is appropriate for undergraduate students or students who are not computer science students. The project will involve researching and understanding the topic in question and formulating a clear bioinformatics problem and suggesting potential tools for it. Finally, you will require to write a term paper on the topic.
    - **Alternative Splicing**
    - **RNAi**
    - **Single Nucleotide Polymorphism (SNPs)**
    - **Recently Published Bioinformatics Tools**
    - **Extremophiles**