BISC/CS 303: Bioinformatics, Spring 2008

Final Project

Proposals due: Wednesday, April 9th
Oral presentations: Wednesday, April 23rd and Wednesday, May 7th
Final reports due: Monday, May 19th

Throughout the semester, assignments have focused on the theory and implementation of various bioinformatics techniques. The assignments will culminate in a final project consisting of both an oral presentation (which accounts for 10% of the final grade) and a written report (which accounts for 30% of the final grade). Details of the final project are provided below.

TEAMS

Two students may work together as a team on a single project. A team may consist of at most two students. In the case of a team project, the size and complexity of the project should be more substantial than what is expected of a typical single-student project. The project ideas listed below may form the basis for team projects, but you should work closely with your instructors during the early phases of the project in order to make sure that the plan for the project is substantial enough to support a team project. The project proposal (see below) for a team should include a plan of work that indicates how the research efforts will be distributed between the team members.

GUIDELINES FOR THE PROJECT PROPOSAL:

You are encouraged to consult with the instructors when deciding on the topic of your final project. A one to two page project proposal is due Wednesday, April 9th. The project proposal should include a description of the specific topics to be covered in the project as well as an explanation of the primary sources to be used for gathering information. In the case of projects that involve computer programming and implementation of computer algorithms, proposals should include a summary of the application that your program will address, an explanation of the program input and output, a description of data sources to be used for evaluating the program, and an outline of the data structures that your implementation will employ. For team projects, the proposal should include a plan of work that indicates how the research efforts will be distributed between the team members. You will receive feedback from the instructors based on your project proposal.

GUIDELINES FOR THE ORAL PRESENTATION:

The presentation will include the highlights of your research that will be explained in more detail in the report. The oral presentation will take place during the last two classes of the semester and should be approximately 15 minutes in length, regardless of whether you are working on your own or as part of a two person team. Your presentation should include 1) sufficient introduction so that the audience can understand the motivations for your project and 2) enough information about your topic that the audience will absorb the "take-home" points of your project. You are encouraged to consult with instructors when preparing your oral presentation.

GUIDELINES FOR THE WRITTEN REPORT:

The written report should describe a carefully researched topic. The report is due on the last day of exam period. It should be at least eight pages in length (two-column single spaced). The eight-page count should include tables and figures that may be interspersed with the text, but the eight-page count should not include references. A minimum of five references is required for the report. The report must include sufficient background information, a logical flow of topics with relevant transitions between topics, and clear figure and/or tables with appropriate legends. You are encouraged to consult with instructors when developing your report.

PROJECT IDEAS:

Below are some ideas for projects and examples of projects that students have investigated in past semesters. You are welcome to choose a project from this list, but you are also welcome to choose a project different from any listed below. If you choose a project outside of the list then you should discuss your idea with an instructor.

1. Students will investigate recent developments in the scientific research community that have a significant bioinformatics component. Information should be gathered from the primary literature as well as on-line sources. Students should significantly extend their knowledge of both the biological and computational components of these papers. In addition, students should present what has been learned, what questions remain, and how bioinformatic tools are being used to address these questions. In addition, students may want to perform their own bioinformatic analyses using data sets from the papers. Some examples of topics for study include, but are certainly not limited to, the following:

- Detailed investigation of your yeast gene and its putative homologs (particularly any human homolog). Relevant topics might include the following:
    - regulation of the gene, including description of the corresponding regulatory networks, transcription factors, and regulatory sites in the genomic sequence
    - expression trends of the gene as determined from gene expression assays
    - phylogenetic analysis of your gene and its putative homologs

- structure and function of the gene product, including visualization of the 3-dimensional structure and identification of important functional elements within the structure such as ligand binding sites or enzyme active sites
- related genes as well as proteins that associate with your gene product
- metabolic pathways to which your gene product contributes
- other interesting features of your gene and its product, such as whether it is implicated in any disease phenotypes, pharmacogenetic information, etc.

- The origins of the human immunodeficiency virus (HIV), such as the phylogenetic relationship of HIV-1, HIV-2, SIV, and other related viruses, differences between HIV subtypes and geographical distribution of the subtypes, etc.

- Recent papers detailing the comparative genomic analyses of 12 distinct *Drosophila* species, such as use of gene prediction algorithms, analysis of genomic synteny, which genes are common to the species and which genes aren't common, common regulatory motifs, etc.

- Comparative genomic analyses and functional evaluation to discern what determines whether a bacterial species behaves as a symbiont or as a pathogen (e.g. *Sinorhizobium meliloti* is a nitrogen-fixing symbiont, but the closely-related bacteria *Brucella abortus* and *Agrobacterium tumefaciens* are pathogens). What genes are required for pathogenesis? Symbiosis? Both? What are the roles of pathogenicity islands, symbiosis islands? etc.

- Pandemic influenza viruses: How does a pandemic flu occur? What genetic changes are required for an avian flu virus to cross the species barrier? What is the origin of the 1918 pandemic flu virus? Milestone 8 was just the tip of the iceberg…

- Comparative genomics in model systems research: for example, how does the human genome compare with the mouse genome? How is this useful in disease research? What are the pros and cons of using the mouse as a model system? It might be useful to investigate one particular human disease for which the mouse is currently used as a model system.

2. Students will study and implement a significant algorithm with applications in genomics or molecular biology. You will be responsible for consulting the literature and on-line sources for background on your algorithm. Your project should consist of the following components.

- an explanation of the problem in genomics or molecular biology that your algorithm addresses
- background information on the design and evolution of the algorithm in the scientific community
- sets of appropriate data on which your algorithm will be evaluated
- a well-documented implementation of your algorithm from scratch
- a summary of the data structures used in the implementation

- an analysis of the algorithm's performance
- a description of extensions to the algorithm

Algorithms and their applications are not restricted to topics covered in class. A few example topics are listed below:

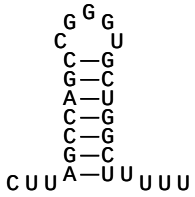
- Clustering algorithms for microarray data. An advanced version of a clustering algorithm could be implemented/investigated or multiple simple clustering algorithms could be implemented and their performance compared. Example clustering algorithms include: hierarchical clustering, *k*-means clustering, a graph-theoretic algorithm such as CAST, self-organizing maps, and model-based clustering algorithms.

- In class, we studied algorithms for BLAST and for pairwise sequence alignment. These algorithms could be investigated in more detail. For example, you could implement the BLAST algorithm in order to compare a query sequence to an entire database of target sequences. You extend the pairwise sequence alignment algorithm to include affine gaps rather than just linear gaps, to print out the optimal alignment, to work on protein sequences, to use different scoring matrices, etc.

- Investigate and implement multiple sequence alignment algorithms, including construction of a guide tree followed by progressive multiple sequence alignment.

- Design an approach for constructing a phylogenetic tree from a set of genomic sequences. The approach might employ distance-based methods or character-based methods, or might compare the two.

- Since RNAs are generally single-stranded, it is often energetically favorable for them to fold into a secondary structure so that nucleotides in one part of the sequence basepair with nucleotides in another part of the sequence. For example, the RNA sequence CUUAGCCAGCCGGGUGCUGGCUUUUU might fold into the following secondary structure, `...((((((((.....))))))))....`:



  A dynamic programming algorithm can be used to identify the optimal secondary structure for a given RNA sequence. You could study and implement this algorithm.

- Investigate algorithms for finding patterns, motifs, and regulatory sites in sets of genomic sequences. For example, given a set of sequences that are believed to be co-regulated,

you could look for common regulatory sites in the sequences using an expectation-maximization approach or using Gibbs sampling or comparing the two approaches.

- Identifying genes in genomic sequences. In Milestone 3, we investigated approaches that identified all ORFs in a genomic sequence as well as those ORFs that have properties suggestive of actual genes. These ORF-finders, or rather gene-finders, can be extended to better identify likely genes in genomic sequences. For example, frequency of amino acids in an ORF can be compared to the expected frequency of amino acids for actual genes, and if the two frequencies are sufficiently similar, the ORF may be hypothesized to be a gene. A gene-finding algorithm can then be tested on various genomic sequences (such as a yeast chromosome) to evaluate how many of the actual yeast genes are accurately predicted by the algorithm.