Computational Biology: Genomes, Networks, Evolution

6.047 / 6.878
Manolis Kellis
James Galagan

TAs: David Sontag / Mike Lin

Goals for the term

• Introduction to computational biology
  – Fundamental problems in computational biology
  – Algorithmic/machine learning techniques for data analysis
  – Research directions for active participation in the field

• Ability to tackle research
  – Problem set questions: algorithmic rigorous thinking
  – Programming assignments: hands-on experience w/ real datasets

• Final project:
  – Research initiative to propose an innovative project
  – Ability to carry out project’s goals, produce deliverables
  – Write-up goals, approach, and findings in conference format
  – Present your project to your peers in conference setting

Course outline

• Organization
  – Duality: Computation and Biology
    • Important biological problems
    • Fundamental computational techniques
  – Foundations and Frontiers
    • First half: well-defined problems and general methodologies
    • Second half: in-depth look at complex problems, combine techniques learned, opens to projects, research directions

• Topics covered
  – First half: the foundations
    • String matching, genome analysis, expression clustering/classification, regulatory motifs, biological networks, evolutionary theory, populations
  – Second half: the frontiers
    • Comparative genomics, Bayesian networks, systems biology, genome assembly, metabolic modeling, miRNA, genome evolution

Books used in the course

Bioinformatics Algorithms
Jones & Pevzner

Introduction to Biocomputing
Durbin et al.

Price: ~$40
Availability: Quantum Books (J), amazon.com (J&D)
MIT libraries: Both books have several copies on reserve

Why Computational Biology?

Administativia

• Course information
  – Lecturers: Manolis Kellis and James Galagan
  – TAs: David Sontag and Mike Lin
  – TR11-12:30, in 3-370
  – http://stellar.mit.edu/S/course/6/fa07/6.047/ or compbio.mit.edu/6.047

• Grading:

<table>
<thead>
<tr>
<th>Problem sets</th>
<th>Final Project</th>
<th>Midterm 25%</th>
<th>Final Exam 10%</th>
</tr>
</thead>
<tbody>
<tr>
<td>40%</td>
<td>25%</td>
<td>20%</td>
<td>5%</td>
</tr>
</tbody>
</table>

• 4 problem sets:
  – Each problem set: 10%, covers 3-4 lectures, contains 3-4 problems.
  – Algorithmic problems and programming assignments
  – Graduate version includes additional problem on current research

• Exams
  – In-class midterm, no final exam

• Collaboration policy
  – Collaboration allowed, but you must:
    • Work independently on each problem before discussing it
    • Write solutions on your own
    • Acknowledge sources and collaborators. No outsourcing.

• Problem set questions: algorithmic rigorous thinking
  – Programming assignments: hands-on experience w/ real datasets

• Final project:
  – Research initiative to propose an innovative project
  – Ability to carry out project’s goals, produce deliverables
  – Write-up goals, approach, and findings in conference format
  – Present your project to your peers in conference setting

• Organization
  – Duality: Computation and Biology
    • Important biological problems
    • Fundamental computational techniques
  – Foundations and Frontiers
    • First half: well-defined problems and general methodologies
    • Second half: in-depth look at complex problems, combine techniques learned, opens to projects, research directions

• Topics covered
  – First half: the foundations
    • String matching, genome analysis, expression clustering/classification, regulatory motifs, biological networks, evolutionary theory, populations
  – Second half: the frontiers
    • Comparative genomics, Bayesian networks, systems biology, genome assembly, metabolic modeling, miRNA, genome evolution

Why Computational Biology?
Why Computational Biology: Last year’s answers

- Lots of data (*lots of data)
- There are rules
- Pattern finding
- It’s all about data
- Ability to visualize
- Simulations
- Guess + verify (generate hypotheses for testing)
- Propose mechanisms / theory to explain observations
- Networks / combinations of variables
- Efficiency (reduce experimental space to cover)
- Informatics infrastructure (ability to combine datasets)
- Correlations
- Life itself is digital: Understand cellular instruction set

Challenges in Computational Biology

1. Genome Assembly
2. Gene Finding
3. Comparative Genomics
4. Evolutionary Theory
5. Regulatory motif discovery
6. Comparative Genomics
7. Database lookup
8. Sequence alignment
9. Gene expression analysis
10. Protein network analysis
11. Cluster discovery
12. Gibbs sampling
13. Metabolic modeling
14. Emerging network properties

Course Outline

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Topic</th>
<th>Homework</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Thu Sep 06</td>
<td>Intro: Biology, Algorithms, Machine Learning</td>
<td>PS1 (due 9/11)</td>
</tr>
<tr>
<td>2</td>
<td>Tue Sep 11</td>
<td>Global/local alignment/DynProg</td>
<td>PS2 (due 9/11)</td>
</tr>
<tr>
<td>3</td>
<td>Thu Sep 13</td>
<td>StringSearch/Blast/DB Search</td>
<td>PS3 (due 9/18)</td>
</tr>
<tr>
<td>4</td>
<td>Tue Sep 18</td>
<td>HMMs1 - Evaluation / Parsing</td>
<td>PS4 (due 10/1)</td>
</tr>
<tr>
<td>5</td>
<td>Thu Sep 20</td>
<td>HMMs2 - Posterior Decoding/Learning</td>
<td>Recitation 3: Posterior decoding review, Baum-Welch Learning (due 10/1)</td>
</tr>
<tr>
<td>6</td>
<td>Tue Sep 25</td>
<td>Conditional Random Fields/Gene Finding</td>
<td>PS5 (due 10/22)</td>
</tr>
<tr>
<td>7</td>
<td>Thu Sep 27</td>
<td>Clustering/Basics/Gene Expression/Sequence Clustering</td>
<td>Recitation 4: Microarrays</td>
</tr>
<tr>
<td>8</td>
<td>Tue Oct 02</td>
<td>Classification/Feature Selection/ROC/SVM</td>
<td>PS6 (due 10/10)</td>
</tr>
<tr>
<td>9</td>
<td>Thu Oct 04</td>
<td>Regulatory Motifs/Gibbs Sampling/EM (due 10/10)</td>
<td>Recitation 5: Entropy, Information, Background models</td>
</tr>
<tr>
<td>10</td>
<td>Tue Oct 09</td>
<td>Biological Networks/Graphs/Network Motifs</td>
<td>PS7 (due 10/22)</td>
</tr>
<tr>
<td>12</td>
<td>Tue Oct 16</td>
<td>Molecular Evolution/Coalescence/Selection/MKS/KaKs</td>
<td>PS8 (due 11/11)</td>
</tr>
<tr>
<td>13</td>
<td>Thu Oct 18</td>
<td>Population Genomics - Fundamentals</td>
<td>PS9 (due 10/22)</td>
</tr>
<tr>
<td>14</td>
<td>Tue Oct 23</td>
<td>Population Genomics - Association Studies</td>
<td>Recitation 7: Population genomics</td>
</tr>
<tr>
<td>15</td>
<td>Thu Oct 25</td>
<td>MIDTERM (Review Session Tue Oct 23rd at 7pm)</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Tue Oct 30</td>
<td>Sequencing Technology and Applications</td>
<td>Project: Phase II</td>
</tr>
<tr>
<td>17</td>
<td>Thu Nov 01</td>
<td>Genome Assembly/EulerGraphs Phase I</td>
<td>Project: Phase I</td>
</tr>
<tr>
<td>18</td>
<td>Tue Nov 06</td>
<td>Comparative Genomics1 - Biological Signal Discovery/Evolutionary Signatures (due 11/11)</td>
<td>Project: Phase I</td>
</tr>
<tr>
<td>19</td>
<td>Thu Nov 08</td>
<td>Comparative Genomics2 - Genome Scale Evolution/Genome Duplication</td>
<td>Project: Phase II</td>
</tr>
<tr>
<td>20</td>
<td>Tue Nov 13</td>
<td>Regulatory Networks/Bayesian Networks</td>
<td>Project: Phase II</td>
</tr>
<tr>
<td>21</td>
<td>Thu Nov 15</td>
<td>Metabolic Modeling 1 - Dynamic systems modeling</td>
<td>Project: Phase III</td>
</tr>
<tr>
<td>22</td>
<td>Tue Nov 20</td>
<td>Metabolic Modeling 2 - Flux balance analysis &amp; metabolic control analysis Phase II</td>
<td>Project: Phase III</td>
</tr>
<tr>
<td>23</td>
<td>Thu Nov 22</td>
<td>Thanksgiving Break (due 11/28)</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Tue Nov 27</td>
<td>Systems Biology - Regulation and metabolism</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>Thu Nov 29</td>
<td>Synthetic Biology - Designing and building biological systems</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>Tue Dec 04</td>
<td>Phylogenomics - Evolution in the context of complete genomes</td>
<td>Project: Phase III</td>
</tr>
<tr>
<td>27</td>
<td>Thu Dec 06</td>
<td>Pre-transcriptional and miRNA regulatory networks</td>
<td>Project: Phase III</td>
</tr>
<tr>
<td>28</td>
<td>Tue Dec 11</td>
<td>Final Presentations - Part I</td>
<td>Project: Phase III</td>
</tr>
</tbody>
</table>

Molecular Biology Primer

“Central dogma” of Molecular Biology

DNA

RNA

Protein

DNA: The double helix

- The most noble molecule of our time
DNA: the molecule of heredity

- Self-complementarity sets molecular basis of heredity
- Knowing one strand, creates a template for the other
- “It has not escaped our notice that the specific pairing we have postulated immediately suggests a possible copying mechanism for the genetic material.” Watson & Crick, 1953

DNA: chemical details

- Bases hidden on the inside
- Phosphodiester backbone
- Weak hydrogen bonds hold the two strands together
- This allows low-energy opening and re-closing of two strands
- Anti-parallel strands
- Extension 5’→3’ triphosphate coming from newly added nucleotide

The only pairings are:
- A with T
- C with G

DNA: deoxyribose sugar

DNA: the four bases

<table>
<thead>
<tr>
<th>The Nucleotides of DNA</th>
<th>Adenine</th>
<th>Thymine</th>
<th>Cytosine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Purine</td>
<td>Purine</td>
<td>Pyrimidine</td>
<td>Pyrimidine</td>
</tr>
<tr>
<td>Weak</td>
<td>Strong</td>
<td>Weak</td>
<td>Strong</td>
</tr>
<tr>
<td>Amino</td>
<td>Keto</td>
<td>Keto</td>
<td>Amino</td>
</tr>
</tbody>
</table>

DNA: base pairs

DNA: sequences

5’-AGAGAG-3’

AGAG or CTCT
DNA packaging

- **Why packaging**
  - DNA is very long
  - Cell is very small

- **Compression**
  - Chromosome is 50,000 times shorter than extended DNA

- **Using the DNA**
  - Before a piece of DNA is used for anything, this compact structure must open locally

Chromosomes inside the cell

- **Eukaryote cell**
- **Prokaryote cell**

"Central dogma" of Molecular Biology

DNA makes RNA

RNA makes Protein

Genes control the making of cell parts

- **The gene is a fundamental unit of inheritance**
  - Each DNA molecule \( \cong 10,000 \) genes
  - 1 gene \( \cong 1 \) functional element (one "part" of cell machinery)
  - Every time a "part" is made, the corresponding gene is:
    - Copied into mRNA, transported, used as blueprint to make protein

- **RNA is a temporary copy**
  - The medium for transporting genetic information from the DNA information repository to the protein-making machinery is an RNA molecule
  - The more parts are needed, the more copies are made
  - Each mRNA only lasts a limited time before degradation

mRNA: The messenger

- **Information changes medium**
  - single strand vs. double strand
  - ribose vs. deoxyribose sugar

From DNA to RNA: Transcription

- **ATACGCTACGT**
- **UAUGCCAUUGCA**

Complementary base-pairing in hybrid

Compatible base-pairing in protein

unacil (RNA) thymine (DNA)
In Eukaryotes, not every part of a gene is coding
- Functional exons interrupted by non-translated introns
- During pre-mRNA maturation, introns are spliced out
- In humans, primary transcript can be $10^6$ bp long

Alternative splicing can yield different exon subsets for the same gene, and hence different protein products.

RNA can be functional
- Single Strand allows complex structure
  - Self-complementary regions form helical stems
  - Three-dimensional structure allows functionality of RNA

Four types of RNA
- mRNA: messenger of genetic information
- tRNA: codon-to-amino acid specificity
- rRNA: core of the ribosome
- snRNA: splicing reactions

To be continued…
- We’ll learn more in a dedicated lecture on RNA world
- Once upon a time, before DNA and protein, RNA did all

RNA structure: Secondary and tertiary

Splicing machinery made of RNA

"Central dogma" of Molecular Biology

Proteins carry out the cell’s chemistry
- More complex polymer
  - Nucleic Acids have 4 building blocks
  - Proteins have 20. Greater versatility
  - Each amino acid has specific properties
- Sequence $\rightarrow$ Structure $\rightarrow$ Function
  - The amino acid sequence determines the three-dimensional fold of protein
  - The protein’s function largely depends on the features of the 3D structure
- Proteins play diverse roles
  - Catalysis, binding, cell structure, signaling, transport, metabolism
Protein structure

Beta-barrel
Some antiparallel b-sheet domains are better described as b-barrels rather than b-sandwiches, for example amyloid plaques and prions. Note that some structures are intermediate between the extreme barrel and sandwich arrangements.

Protein building blocks

• Amino Acids

From RNA to protein: Translation

The Genetic Code

• Degeneracy of the genetic code
  – To encode 20 amino acids, two nucleotides are not enough \(4^2=16\). Three nucleotides are too many \(4^3=64\)
  – The genetic code is degenerate. Same amino acid can be represented by more than one codon. Room for innovation
  – Moreover, amino acids with similar properties can be substituted for each other without changing the structure of the protein

• Six possible translation frames for every nucleotide stretch
  – GCC - GCU - GUA - CGA - AUA \(\rightarrow\) Ala - Cys - Leu - Arg - Bc
  – C.CU.LGU - UAC - GAA - UUA \(\rightarrow\) Leu - Val - Tyr - Glu - Leu
  – Stop codon every 3/64. Long ORFs are unlikely, probably genes
  – In some viruses as many as four overlapping frames are functional

Summary: The Central Dogma

DNA makes RNA makes Protein