CSE 584A Class 1

Jeremy Buhler

January 20, 2016

1 Introduction

Welcome to CSE 584!

- For anyone who doesn't know me, I'm Jeremy Buhler (jbuhler@wustl.edu)
- Please note the course web site, http://classes.engineering.wustl.edu/cse584a. This is the source for announcements, assignments, handouts, and whatnot.
- **No required course text**. We'll use my notes and research papers as our source material.
- The *Cartoon Guide to Genetics* is an optional text, available from Amazon for $15 plus shipping.
- You may also be interested in Gusfield's *Algortihms on Strings, Trees, and Sequences*, available from Amazon for $96 plus shipping.

Before I continue, I need to know who you are.

- Who is actually signed up for the course?
- How many grads? How many undergrads?
- Who here has had some biology training since high school?
- Who has taken CSE 587 / BIO 5495 / BME 537?
- Who has seen or used the Smith-Waterman algorithm? Suffix trees/arrays? BLAST? Short-read aligners (Bowtie, BWA, others)?

Before we get into fun stuff, we need to cover some administrivia.

- **Assignments**: 4 homeworks, plus a final project
- The homeworks will be mostly written, with some programming. They're intended to make you think about the algorithms we discuss in class.
- We'll use electronic submission - Blackboard for written homeworks, SVN repositories for code. More details on this and on grading when the first homework goes out.

- The final project will be a substantial implementation of an application of your choosing, preferably a useful biosequence comparison task. (You'll have about a month to do it; it should be ambitious enough to take a month.)
- **Collaboration**: I have a policy, which I hope is not too draconian. It's on the web site – please read it!

2 On to the Good Stuff – Biology

What is this course about, anyway?

- The CSE answer: selected topics in **stringology** (study of the properties of strings and of algorithms for searching/manipulating them)
- What's a string, anyway? Let Σ be an alphabet of characters. A string s over Σ is an ordered array of characters from Σ. Denote the ith character of s as $s[i]$. We often write $s \in \Sigma^*$ to mean "s is a string over Σ."
- Stringology is a huge field with ties to databases and algorithms. Examples: searching web page database; scanning security logs; building an index or concordance for a book; computing anagrams of a word; solving a crossword puzzle
- What does this have to do with bio? Detecting *similarity* between biological sequences (DNA or protein, special kinds of strings)
- (What do I mean by "similarity"? Why does it matter, and how is it related to "conservation"? More in a bit...)

I suspect this audience has a pretty mixed bio background. If the following is not news to you, please take a nap until I'm done.

- We'll talk mostly about DNA in here, but also about protein and larger alphabets.
- DNA is abstractly a string of *nucleotides* or *bases* from the alphabet A, C, G, T.
- DNA exists in nature as a *double-stranded molecule*, in which bases pair according to the rule $A - T$ and $G - C$. These pairs are *complementary*.
- Given a DNA sequence (string) s, its *reverse complement* is the string obtained by replacing each base $s[i]$ with its complementary base and reversing the result.
- (Complementarity enables DNA to replicate itself.)
- Some DNA encodes functional products, such as enzymes or structural proteins. The pieces of DNA that code for these products are the *genes*, which make up less than 2% of the human *genome*.

- A gene's DNA is *transcribed* into RNA, which has the same sequence except that each T (thymine) base is replaced by a U (uracil).
- RNA for protein-coding genes is *translated* into *protein*, which is a string over the alphabet of 20 *amino acids*.
- Each group of three DNA/RNA bases forms a *codon*, which corresponds to one amino acid. (Some codons indicate that translation should stop.)

- Codons are degenerate – several of them may encode the same amino acid. Much of the time, the third base of a codon is not important in determining the encoded amino acid. (E.g., "CG*" codes for arginine)

For a detailed discussion of the structure and function of DNA and the "Central Dogma," see Gonick and Wheelis, or check out http://www.dnaftb.org/ .

3 Why Apply Stringology to Biosequences?

Why are DNA, RNA, and protein sequences common targets for stringology?

- Biologists are remarkably good at sequencing DNA.
- Since the mid-1980s, there has been rapid development in *sequencing machines* that can read a DNA (or RNA) sequence.
- As a result, we have accumulated vast databases of known sequence.
- For more information on what kinds of databases (not just sequence) are publicly available, see, e.g.,
  - `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4702911/`
  - `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC4702806/`
  - `http://europepmc.org/articles/PMC4702932`

What do we do with all this sequence?

- **Map Back**: if sequences came from a known source, determine where in the source they occur.
- *Applications*: sequence expressed mRNA from a known genome, then figure out which genes it is from (RNASeq). Sequence genomic DNA that is special in some way (e.g. binds protein, is methylated ...), and figure out where in the genome it came from (ChipSEQ, bisulfite sequencing)
- **Simplify**: sequencers typically yield many redundant, overlapping copies of the same sequence.

- Need to collapse these into fewer sequences by detecting and removing duplicates, merging overlapping sequences (assembly).
- **Cluster**: can we group non-identical but similar sequences?
- Similarity (up to evolutionary changes, polymorphisms, or sequencing error) may be useful in relating sequences.
- *Examples*: same gene in several species; related genes all in same family or with same functional domain; retroviral and repeat remnants
- *Applications*: phylogeny building, discovering gene families in metagenomic data, whole-genome alignments between species
- **Correlate**: where does a particular sequence occur?
- **Example**: sequence DNA in blood samples from many people with a common infectious disease.
- Look for non-human sequence that is common to all samples and not present in healthy people.
- Result may identify a new infections agent (e.g. a virus).

4 What Makes Stringology on Biosequences (Computationally) Interesting?

- Sizes of databases and data sets used for comparison are intimidating.
- (Throughout, we'll emphasize search algorithms that make *efficient* use of space and time.)
- Sufficiently similar sequences often contain regions where the bases/amino acids match each other perfectly.
- Therefore, we'll spend much time on how to identify such *exact* matches quickly.
- The most effective strategies for large data sets entail building *indices* to accelerate search.
- We'll start with pretty simple ways to index a large text database, then work our way up to more complex approaches.
- (It's amazing what you can do in linear time and space with a small constant factor!)
- However, even sequences that have been maintained without many changes over time are rarely identical over their entire lengths, so we will also talk about how to find *inexact* matches.
- *Important question*: how do we measure similarity?
- Biologists have definite ideas about how sequences evolve, so a correct measure of similarity between sequences may not be just the number of differing characters.

- Example: transitions (A-G and C-T) versus transversions (everything else)

- We'll talk about how to accommodate these ideas in the section on dynamic programming alignment.
- The end-game is to fuse awesome data structures for exact match with biologically meaningful ideas of inexact matching.
- Doing this effectively at scale is the cutting edge of sequence comparison today.

I can't be comprehensive in my coverage of methods, but I can show you some of the basic tools, along with elaborations you might not have considered. My goal is to equip you to be a productive user of the latest technology in this area, and to have a starting point if you want to get into research about it.