

Class 13: Gene Prediction (1)

Held: Thursday, 6 October 2011

Summary: We consider ways in which we can computationally identify genes (or sequences likely to be genes) in longer DNA sequences.

Related Pages:

- EBoard.
- Reading: Chapter 6 ; Kellis et al. 2003.

Notes:

- Response to Kellis *et al.* is due Tuesday.
- Tonight is homecoming in Grinnell. If you've never seen a small-town Iowa homecoming parade, it's worth going at least once.
- If you must partake of 10/10 this weekend, please partake responsibly.
- Today's programming lab is your next programming assignment, and is due Thursday.
- EC for Today's CS extra on Computer Vision (4:30, 3821).
- EC for Friday's Biology Seminar (noon, 2021).
- EC for Healthy Iowa Walk noon on Friday.
- EC for Friday's Volleyball game (7pm, Darby).
- EC for Saturday's Football game (1 pm).
- EC for Saturday's Men's Soccer game (1:30 pm).
- EC for Orchestra, Saturday (2pm, S-L).

Overview:

- Quick review: Shortest Superstring.
- Check in: What is a gene?
- Gene Prediction: Going beyond DNA.
- Strategies for predicting useful ORFs.
- Obstacles to finding genes.
- Detour: Why Sam loves this topic.
- Web exploration.
- Programming exploration.

The Shortest Superstring Problem, Revisited

- Note that we effectively designed two algorithms that approximate the shortest superstring in the previous class.
- Algorithm 1

```

Algorithm
  ShortSuperString(S)
Input
  S, a set of strings
Output
  super, a single string
  Every string in S appears somewhere in super
Begin
  Let super be a largest element of S.
  Remove super from S.
  while strings remain in S
    find s, the string in S that best aligns with super
    remove s from S
    super = align(super, s)
  return super
End

```

- Algorithm 2

```

Begin
  while S contains more than one string
    pick, s1 and s2, the two strings in S that best align
    remove s1 and s2 from S
    add align(s1,s2) to S
  return S[0]
End

```

- *Example needed*

Gene Prediction: Going Beyond DNA Sequences

- We've now figured out how to get long sequences of DNA.
 - See previous classes for details
- What can we do with those sequences?
 - If we sequence something new, we can *search* those sequences for a similar sequence.
 - We can *compare* related sequences to see how they change (e.g., to build trees)
 - ...
- One clearly useful thing would be to find the genes in the genome
 - But it's a very hard problem

Review: What is a Gene?

Intentionally left nearly blank.

Strategies for Gene Prediction

- Match with existing genes in other organisms
 - Feels like a bit of a chicken-and-the-egg problem
- String matching
 - Computer science is simple; we're good at finding strings (even approximate strings)

- Biology is messy: We don't know what strings to predict.
- Other characteristics, such as base-pair frequency

Obstacles to Gene Prediction

- Incomplete knowledge: Promotor sequences not universal (and probably not always known)
- Introns and Exons
- ...

Why Sam Loves Gene Prediction

- This is where we get cool opportunities in bioinformatics
- Open problems
- Need biological expertise to hypothesize “patterns” that indicate genes
- Need computational expertise to turn those patterns in to code.
- Need biological expertise to analyze the results.

Web Exploration

- Work on the Web exploration for Chapter 6.

Project 6.5

- Start work on the programming project for Chapter 6.
- We'd like you to turn in a finished version by next Thursday.

Copyright © 2009-2011 Vida Praitis and Samuel A. Rebelsky. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.