Computational Approaches to Gene Finding

EE381V Lecture #26
Outline

1. Basic Information and Introduction

2. Some Mathematical Concepts and Definitions

3. Examples of Gene Finding
1. Basic Information

- What types of predictions can we make?
- What are they based on?
Bioinformatics as *Extrapolation*

- Computational gene finding is a process of:
  - Identifying common phenomena in known genes
  - Building a computational framework/model that can accurately describe the common phenomena
  - Using the model to scan uncharacterized sequence to identify regions that match the model, which become putative genes
  - Test and validate the predictions
Different Types of Gene Finding

- **Protein coding genes**
  - Prokaryotic
    - No introns, simpler regulatory features
  - Eukaryotic
    - Exon-intron structure
    - Complex regulatory features
Approaches to Gene Finding

- **Direct**
  - Exact or near-exact matches of, e.g., cDNA or Proteins from the same, or closely related organism

- **Indirect**
  1. Look for something that looks like one gene (*homology*)
  2. Look for something that looks like all genes (*ab initio*)
  3. Hybrid, combining homology and ab initio (and perhaps even direct) methods
Pieces of a (Eukaryotic) Gene

- Promoter (~10^3 bp)
- Enhancers (~10^1-10^2 bp)
- Exons (~10^2-10^3 bp)
- Introns (~10^2-10^5 bp)
- Other regulatory sequences (~10^1-10^2 bp)
What is it about genes that we can measure (and model)?

- Most of our knowledge is biased towards protein-coding characteristics
  - ORF (Open Reading Frame): a DNA sequence that does not contain a stop codon in a given reading frame.
  - Codon Usage: most frequently measured by CAI (Codon Adaptation Index)

- Other phenomena
  - Nucleotide frequencies and correlations:
    - value and structure
  - Functional sites:
    - splice sites, promoters, etc.
A simple measure: ORF length Comparison of Annotation and Spurious ORFs in *S. cerevisiae*
Codon Adaptation Index (CAI)

\[ \text{CAI} = \prod_{i=\text{codons}} \left[ \frac{f_{\text{codon}_i}}{f(\text{codon}_i)_{\text{max}}} \right] \]

- Parameters are empirically determined by examining a "large" set of example genes
- This is not perfect
  - Genes sometimes have unusual codons for a reason
  - The predictive power is dependent on length of sequence
## CAI Example

### Counts per 1000 codons

<table>
<thead>
<tr>
<th>Codon</th>
<th>Amino Acid</th>
<th>hs</th>
<th>sc</th>
<th>Codon</th>
<th>Amino Acid</th>
<th>hs</th>
<th>sc</th>
<th>Codon</th>
<th>Amino Acid</th>
<th>hs</th>
<th>sc</th>
<th>Codon</th>
<th>Amino Acid</th>
<th>hs</th>
<th>sc</th>
</tr>
</thead>
<tbody>
<tr>
<td>UUU</td>
<td>Phe</td>
<td>16.6</td>
<td>26.0</td>
<td>UCU</td>
<td>Ser</td>
<td>14.5</td>
<td>23.6</td>
<td>UAU</td>
<td>Tyr</td>
<td>12.1</td>
<td>18.8</td>
<td>UGU</td>
<td>Cys</td>
<td>9.7</td>
<td></td>
</tr>
<tr>
<td>UUC</td>
<td>Phe</td>
<td>20.7</td>
<td>18.2</td>
<td>UCC</td>
<td>Ser</td>
<td>17.7</td>
<td>14.2</td>
<td>UAC</td>
<td>Tyr</td>
<td>16.3</td>
<td>14.7</td>
<td>UGC</td>
<td>Cys</td>
<td>12.4</td>
<td></td>
</tr>
<tr>
<td>UUA</td>
<td>Leu</td>
<td>7.0</td>
<td>26.3</td>
<td>UCA</td>
<td>Ser</td>
<td>11.4</td>
<td>18.8</td>
<td>UAA</td>
<td>stop</td>
<td>0.7</td>
<td>1.0</td>
<td>UGA</td>
<td>stop</td>
<td>1.3</td>
<td></td>
</tr>
<tr>
<td>UUG</td>
<td>Leu</td>
<td>12.0</td>
<td>27.1</td>
<td>UCG</td>
<td>Ser</td>
<td>4.5</td>
<td>8.6</td>
<td>UAG</td>
<td>stop</td>
<td>0.5</td>
<td>0.5</td>
<td>UGG</td>
<td>Trp</td>
<td>13.0</td>
<td></td>
</tr>
<tr>
<td>CUU</td>
<td>Leu</td>
<td>12.4</td>
<td>12.2</td>
<td>CCU</td>
<td>Pro</td>
<td>17.2</td>
<td>13.6</td>
<td>CAU</td>
<td>His</td>
<td>10.1</td>
<td>13.7</td>
<td>CGU</td>
<td>Arg</td>
<td>4.7</td>
<td></td>
</tr>
<tr>
<td>CUC</td>
<td>Leu</td>
<td>19.3</td>
<td>5.4</td>
<td>CCC</td>
<td>Pro</td>
<td>20.3</td>
<td>6.8</td>
<td>CAC</td>
<td>His</td>
<td>14.9</td>
<td>7.8</td>
<td>CGC</td>
<td>Arg</td>
<td>11.0</td>
<td></td>
</tr>
<tr>
<td>CUA</td>
<td>Leu</td>
<td>6.8</td>
<td>13.4</td>
<td>CCA</td>
<td>Pro</td>
<td>16.5</td>
<td>18.2</td>
<td>CAA</td>
<td>Gln</td>
<td>11.8</td>
<td>27.5</td>
<td>CGA</td>
<td>Arg</td>
<td>6.2</td>
<td></td>
</tr>
<tr>
<td>CUG</td>
<td>Leu</td>
<td>40.0</td>
<td>10.4</td>
<td>CCG</td>
<td>Pro</td>
<td>7.1</td>
<td>5.3</td>
<td>CAG</td>
<td>Gln</td>
<td>34.4</td>
<td>12.2</td>
<td>CGG</td>
<td>Arg</td>
<td>11.6</td>
<td></td>
</tr>
<tr>
<td>AUU</td>
<td>Ile</td>
<td>15.7</td>
<td>30.2</td>
<td>ACU</td>
<td>Thr</td>
<td>12.7</td>
<td>20.2</td>
<td>AAU</td>
<td>Asn</td>
<td>16.8</td>
<td>36.0</td>
<td>AGU</td>
<td>Ser</td>
<td>11.7</td>
<td></td>
</tr>
<tr>
<td>AUC</td>
<td>Ile</td>
<td>22.3</td>
<td>17.1</td>
<td>ACC</td>
<td>Thr</td>
<td>19.9</td>
<td>12.6</td>
<td>AAC</td>
<td>Asn</td>
<td>20.2</td>
<td>24.9</td>
<td>AGC</td>
<td>Ser</td>
<td>19.3</td>
<td></td>
</tr>
<tr>
<td>AUA</td>
<td>Ile</td>
<td>7.0</td>
<td>17.8</td>
<td>ACA</td>
<td>Thr</td>
<td>14.7</td>
<td>17.7</td>
<td>AAA</td>
<td>Lys</td>
<td>23.6</td>
<td>42.1</td>
<td>AGA</td>
<td>Arg</td>
<td>11.2</td>
<td></td>
</tr>
<tr>
<td>AUG</td>
<td>Met</td>
<td>22.2</td>
<td>20.9</td>
<td>ACG</td>
<td>Thr</td>
<td>6.4</td>
<td>8.0</td>
<td>AAG</td>
<td>Lys</td>
<td>33.2</td>
<td>30.8</td>
<td>AGG</td>
<td>Arg</td>
<td>11.1</td>
<td></td>
</tr>
<tr>
<td>GUU</td>
<td>Val</td>
<td>10.7</td>
<td>22.0</td>
<td>GCU</td>
<td>Ala</td>
<td>18.4</td>
<td>21.1</td>
<td>GAU</td>
<td>Asp</td>
<td>22.2</td>
<td>37.8</td>
<td>GGU</td>
<td>Gly</td>
<td>10.9</td>
<td></td>
</tr>
<tr>
<td>GUC</td>
<td>Val</td>
<td>14.8</td>
<td>11.6</td>
<td>GCC</td>
<td>Ala</td>
<td>28.6</td>
<td>12.6</td>
<td>GAC</td>
<td>Asp</td>
<td>26.5</td>
<td>20.4</td>
<td>GGC</td>
<td>Gly</td>
<td>23.1</td>
<td></td>
</tr>
<tr>
<td>GUA</td>
<td>Val</td>
<td>6.8</td>
<td>11.7</td>
<td>GCA</td>
<td>Ala</td>
<td>15.6</td>
<td>16.2</td>
<td>GAA</td>
<td>Glu</td>
<td>28.6</td>
<td>45.9</td>
<td>GGA</td>
<td>Gly</td>
<td>16.4</td>
<td></td>
</tr>
<tr>
<td>GUG</td>
<td>Val</td>
<td>29.3</td>
<td>10.7</td>
<td>GCG</td>
<td>Ala</td>
<td>7.7</td>
<td>6.1</td>
<td>GAG</td>
<td>Glu</td>
<td>40.6</td>
<td>19.1</td>
<td>GGG</td>
<td>Gly</td>
<td>16.5</td>
<td></td>
</tr>
</tbody>
</table>
General Things to Remember about (Protein-coding) Gene Prediction Software

- It is, in general, organism-specific

- It works best on genes that are *reasonably* similar to something seen previously

- It finds protein coding regions far better than non-coding regions

- In the absence of external (direct) information, alternative forms will not be identified

- It is imperfect! (It’s biology, after all…)
2. Some Mathematical Concepts and Definitions

- Models

- Bayesian Statistics

- Markov Models & Hidden Markov Models
In gene finding, models can best be thought of as “sequence generators” (e.g., Hidden Markov Models) or “sequence classifiers” (e.g., Neural Networks).

The better (and usually more complex) a model is, the better the performance is likely to be.
Assessing performance: Sensitivity and Specificity

- Testing of predictions is performed on sequences where the gene structure is known

- **Sensitivity** is the fraction of known genes (or bases or exons) correctly predicted
  - “Am I finding the things that I’m supposed to find”

- **Specificity** is the fraction of predicted genes (or bases or exons) that correspond to true genes
  - “What fraction of my predictions are true?”

- In general, increasing one decreases the other
Graphic View of Specificity and Sensitivity

Sn = \frac{TruePositive}{AllTrue} = \frac{TruePositive}{TruePositive + FalseNegative}

Sp = \frac{TruePositive}{AllPositive} = \frac{TruePositive}{TruePositive + FalsePositive}
Quantifying the tradeoff: Correlation Coefficient

\[
CC = \frac{[(TP)(TN) - (FP)(FN)]}{\sqrt{(AN)(PP)(AP)(PN)}}
\]

\[
AN = TN + FP; \ AP = TP + FN;
\]

\[
PP = TP + FP; \ PN = TN + FN
\]
Specificity/Sensitivity Tradeoffs

- Ideal Distribution of Scores

- More Realistically…
Bayesian Statistics

- Bayes’ Rule is at the heart of much predictive software

- In the simplest example, we can simply compare two models, and reduce it to a log-odds ratio

\[
\log \frac{P(M_1|data)}{P(M_2|data)} = \log \frac{P(data|M_1)}{P(data|M_2)} + \log \frac{P(M_1)}{P(M_2)}
\]
Sequence Generation Models: Markov Chains

- A Markov chain is a model for stochastic generation of sequential phenomena.

- Every position in a chain is equivalent.

- The order of the Markov chain is the number of previous positions on which the current position depends.
  - e.g., in nucleic acid sequence, 0-order is mononucleotide, 1\textsuperscript{st}-order is dinucleotide, 2\textsuperscript{nd}-order is trinucleotide, etc.

- The model parameters are the frequencies of the elements at each position (possibly as a function of preceding elements).
MC Models of Sequence Generation

\[ s = ttacggt \ldots \]

- **0\textsuperscript{th}-order**
  \[
P_0(s) = p(t) \cdot p(t) \cdot p(a) \cdot p(c) \cdot p(g) \ldots = \prod_{i=1}^{N} p(s_i)
\]

- **1\textsuperscript{st}-order**
  \[
P_1(s) = p(t) \cdot p(t \mid t) \cdot p(a \mid t) \cdot p(c \mid a) \ldots = p(s_1) \cdot \prod_{i=2}^{N} p(s_i \mid s_{i-1})
\]

- **2\textsuperscript{nd}-order**
  \[
P_2(s) = p(tt) \cdot p(a \mid tt) \cdot p(c \mid ta) \cdot p(g \mid ac) \ldots = p(s_1 s_2) \cdot \prod_{i=3}^{N} p(s_i \mid s_{i-2} s_{i-1})
\]
Hidden Markov Models

- In general, sequences are not monolithic, but can be made up of discrete segments

- Hidden Markov Models (HMMs) allow us to model complex sequences, in which the character emission probabilities depend upon the state

- Think of an HMM as a probabilistic or stochastic sequence generator, and what is hidden is the current state of the model
HMM Details

- An HMM is completely defined by its:
  - State-to-state transition matrix (Φ)
  - Emission matrix (H)
  - State vector (x)

- We want to determine the probability of any specific (query) sequence having been generated by the model; with multiple models, we then use Bayes’ rule to determine the best model for the sequence

- Two algorithms are typically used for the likelihood calculation:
  - Viterbi
  - Forward

- Models are trained with **known examples**
The HMM Matrixes: $\Phi$ and $H$

\[ \Phi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.998 & 0.002 & 0 \\ 0.5 & 0.001 & 0.996 & 0 \\ 0 & 0.001 & 0.002 & 0 \end{bmatrix} \]

\[ H = \begin{bmatrix} 0.28 & 0.32 \\ 0.22 & 0.18 \\ 0.25 & 0.18 \\ 0.25 & 0.32 \end{bmatrix} \]
A more realistic (and complex) HMM model for Gene Prediction (Genie)
Scoring an HMM: Viterbi, Forward, and Forward-Backward

- Two algorithms are typically used for the likelihood calculation: **Viterbi** and **Forward**
  - **Viterbi** is an approximation;
    - The probability of the sequence is determined by using the most likely mapping of the sequence to the model
    - in many cases good enough (gene finding, e.g.), but not always
  - **Forward** is the rigorous calculation;
    - The probability of the sequence is determined by summing over all mappings of the sequence to the model
  - **Forward-Backward** produces a probabilistic map of the model to the sequence
Eukaryotic Gene Prediction: GRAIL II: Neural network based prediction
Open Challenges in Predicting Eukaryotic (Protein-Coding) Genes

- Alternative Processing of Transcripts
  - Splice variants, Start/stop variants
- Overlapping Genes
  - Coding is possible
- Non-canonical functional elements
  - Splice w/o GT-AG
- Small (mini) exons
Open Challenges in Predicting Prokaryotic (Protein-Coding) Genes

- **Start site prediction**
  - Most algorithms are greedy, taking the largest ORF
- **Overlapping Genes**
  - This can be very problematic, esp. with use of Viterbi-like algorithms
- **Non-canonical coding**
Tools for Gene Finding Based on Direct or Homology Evidence

- **BLAST family, FASTA, etc.**
  - Pros: fast, statistically well founded
  - Cons: no understanding/model of gene structure

- **BLAT, Sim4, EST_GENOME, etc.**
  - Pros: gene structure is incorporated
  - Cons: non-canonical splicing, slower than blast
Eukaryotic gene prediction tools and web servers

- Genscan (ab initio), GenomeScan (hybrid)
  - [http://genes.mit.edu/](http://genes.mit.edu/)
- Twinscan (hybrid)
  - [http://genes.cs.wustl.edu/](http://genes.cs.wustl.edu/)
- FGENESH (ab initio)
- GeneMark.hmm (ab initio)
  - [http://opal.biology.gatech.edu/GeneMark/eukhmm.cgi](http://opal.biology.gatech.edu/GeneMark/eukhmm.cgi)
- MZEF (ab initio)
- GraileXP (hybrid)
- GeneID (hybrid)
  - [http://www.l.imim.es/geneid.html](http://www.l.imim.es/geneid.html)
Prokaryotic Gene Prediction

- Glimmer
  - [http://www.tigr.org/~salzberg/glimmer.html](http://www.tigr.org/~salzberg/glimmer.html)
- GeneMark
  - [http://opal.biology.gatech.edu/GeneMark/gmhmm2_prok.cgi](http://opal.biology.gatech.edu/GeneMark/gmhmm2_prok.cgi)
- Critica
- ORNL Annotation Pipeline
Non-protein Coding Gene Tools and Information

- tRNA
  - tRNA-ScanSE
  - FASTRNA
    - [http://bioweb.pasteur.fr/seqanal/interfaces/fastrna.html](http://bioweb.pasteur.fr/seqanal/interfaces/fastrna.html)

- snoRNA
  - snoRNA database
    - [http://rna.wustl.edu/snoRNAdb/](http://rna.wustl.edu/snoRNAdb/)

- microRNA
  - Sfold
    - [http://www.bioinfo.rpi.edu/applications/sfold/index.pl](http://www.bioinfo.rpi.edu/applications/sfold/index.pl)
  - SIRNA
    - [http://bioweb.pasteur.fr/seqanal/interfaces/sirna.html](http://bioweb.pasteur.fr/seqanal/interfaces/sirna.html)