Outline

- Gene structure basics
- Gene finding overview
- GRAIL
- Indel & frame-shift in coding regions

Gene Structure Basics

A brief refresher

Some slides here are "borrowed" from Ken Sung.

Gene

- A gene is a sequence of DNA that encodes a protein or an RNA molecule
- About 30,000 – 35,000 (protein-coding) genes in human genome
- For gene that encodes protein
  - In Prokaryotic genome, one gene corresponds to one protein
  - In Eukaryotic genome, one gene can corresponds to more than one protein because of the process "alternative splicing"

Introns and Exons

- Eukaryotic genes contain introns & exons
  - Introns are seq that are ultimately spliced out of mRNA
  - Introns normally satisfy GT-AG rule, viz. begin w/ GT & end w/ AG
  - Each gene can have many introns & each intron can have thousands bases
- Intron can be very long
- An extreme example is a gene associated with cystic fibrosis in human:
  - Length of 24 introns ~1Mb
  - Length of exons ~1kb

Unlike eukaryotic genes, a prokaryotic gene typically consists of only one contiguous coding region
Reading Frame

- Each DNA segment has six possible reading frames

Forward strand:

<table>
<thead>
<tr>
<th>Reading frame #1</th>
<th>Reading frame #2</th>
<th>Reading frame #3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATG</td>
<td>TGG</td>
<td>GGC</td>
</tr>
<tr>
<td>GCT</td>
<td>TTA</td>
<td>TCA</td>
</tr>
<tr>
<td>TAC</td>
<td>ACG</td>
<td>TCC</td>
</tr>
<tr>
<td>AGC</td>
<td>TGT</td>
<td>AGA</td>
</tr>
<tr>
<td>TCG</td>
<td>GA</td>
<td>A</td>
</tr>
</tbody>
</table>

Reverse strand:

<table>
<thead>
<tr>
<th>Reading frame #4</th>
<th>Reading frame #5</th>
<th>Reading frame #6</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCA</td>
<td>CAA</td>
<td>AAG</td>
</tr>
<tr>
<td>AGC</td>
<td>GGG</td>
<td>GCT</td>
</tr>
<tr>
<td>GTA</td>
<td>TAA</td>
<td>AAG</td>
</tr>
<tr>
<td>AGC</td>
<td>GCC</td>
<td>GCA</td>
</tr>
<tr>
<td>CAT</td>
<td>AT</td>
<td>T</td>
</tr>
</tbody>
</table>

Open Reading Frame (ORF)

- ORF is a segment of DNA with a start codon and an in-frame stop codon at the two ends and no in-frame stop codon in the middle

Start

ORF

Stop

- Each ORF has a fixed reading frame

NB: Other definitions are also used. Most important aspect is that there is no stop codon in the middle.

Coding Region

- Each coding region (exon or whole gene) has a fixed translation frame
- A coding region always sits inside an ORF of same reading frame
- All exons of a gene are on the same strand
- Neighboring exons of a gene could have different reading frames

Frame Consistency

- Neighboring exons of a gene should be frame-consistent

Exercise: Define frame consistency mathematically

What is Gene Finding?

- Find all coding regions from a stretch of DNA sequence, and construct gene structures from the identified exons
- Can be decomposed into
  - Find coding potential of a region in a frame
  - Find boundaries between coding & non-coding regions

Image credit: Xu
Approaches

- Search-by-signal: find genes by identifying the sequence signals involved in gene expression
- Search-by-content: find genes by statistical properties that distinguish protein coding DNA from non-coding DNA
- Search-by-homology: find genes by homology (after translation) to proteins
- State-of-the-art systems for gene finding usually combine these strategies

Relevant Signals for Search-by-Signals

- Transcription initiation
  - Promoter
- Transcription termination
  - Terminators
- Translation initiation
  - Ribosome binding sites
  - Initiation codons
- Translation termination
  - Stop codons
- RNA processing
  - Splice junction

How Search-by-Signal Works

- There are 2 impt regions in a promoter seq
  -10 region, ~10bp before TSS
  -35 region, ~35bp before TSS
- Consensus for –10 region in E. coli is TATAAT, but few promoters actually have this seq
- Recognize promoters by
  - weight matrices
  - probabilistic models
  - neural networks, ...

How Search-by-Content Works

- Encoding a protein affects stats properties of a DNA seq
  - some amino acids used more frequently
  - diff number of codons for diff amino acids
  - for given protein, usually one codon is used more frequently than others
  - Estimate prob that a given region of seq was “caused by” its being a coding seq

How Search-by-Homology Works

- Translate DNA seq in all reading frames
- Search against protein db
- High-scoring matches suggest presence of homologous genes in DNA
  - You can use BLASTX for this

Search-by-Content Example: Codon Usage Method

- Staden & McLachlan, 1982
- Process a seq w/ “window” of length L
- Assume seq falls into one of 7 categories, viz.
  - Coding in frame 0, frame 1, ..., frame 5
  - Non-coding
- Use Bayes’ rule to determine prob of each category
- Assign seq to category w/ max prob
Pr(coding) is the same for each frame if window size fits same number of codons in each frame

• Otherwise, consider relative number of codons in window in each frame

Search-by-Homology Example:
Gene Finding Using BLAST
• High seq similarity typically implies homologous genes
⇒ Search for genes in yeast seq using BLAST
⇒ Extract Feature for gene identification

• Searching all ORFs against known genes in nr db helps identify an initial set of (possibly incomplete) genes
A (yeast) gene starts with ATG and ends with a stop codon, in the same reading frame of ORF. Have "strong" coding potentials, measured by preference models, Markov chain model, ... Have "strong" translation start signal, measured by weight matrix model, ... Have distributions with length, G+C composition, ... Have special seq signals in flanking regions, ...

Coding Signal

- Freq distribution of dimers in protein seq
- E.g., Shewanella
  - Ave freq is 5%
  - Some amino acids prefer to be next to each other
  - Some amino acids prefer to be not next to each other

Exercise: What is shewanella?

Why Dicodon (6-mer)?

- Codon (3-mer)-based models are not as info rich as dicodon-based models
- Tricodon (9-mer)-based models need too many data points

There are
- 4^2 = 64 codons
- 4^4 = 4096 dicodons
- 4^6 = 262144 tricodons.

To make stats reliable, need ~15 occurrences of each X-mer

=> For tricodon-based models, need at least 15*262144 = 3932160 coding bases in our training data, which is probably not going to be available for most genomes

GRAIL, An Important Gene Finding Program

Signals assoc w/ coding regions

Models for coding regions

Signals assoc w/ boundaries

Models for boundaries

Other factors & information fusion

Exercise: In human genome, freq of dicodon "AAAAA" is ~1% in coding region vs ~5% in non-coding region. If you see a region with many "AAAAA", would you guess it is a coding or non-coding region?

Coding Signal

- Dimer preference implies dicodon (6-mers like AAA TTT) bias in coding vs non-coding regions
- Relative freq of a dicodon in coding vs non-coding
  - Freq of dicodon X (e.g., AAA AAA) in coding region = total number of occurrences of X divided by total number of dicodon occurrences
  - Freq of dicodon X (e.g., AAA AAA) in non-coding region = total number of occurrences of X divided by total number of dicodon occurrences

Exercise: In human genome, freq of dicodon "AAAAA" is ~1% in coding region vs ~5% in non-coding region. If you see a region with many "AAAAA", would you guess it is a coding or non-coding region?

Coding Signal

- Most dicodons show bias toward either coding or non-coding regions
  - Foundation for coding region identification

Regions consisting of dicodons that mostly tend to be in coding regions are probably coding regions; otherwise non-coding regions

=> Dicodon freq are key signal used for coding region detection; all gene finding programs use this info
Coding Signal

- Dicodon freq in coding vs non-coding are genome-dependent

Shewanella

Bovine

Dicodon Preference Model

- The preference value $P(X)$ of a dicodon $X$ is defined as
  \[ P(X) = \log \frac{FC(X)}{FN(X)} \]
  where
  - $FC(X)$ is freq of $X$ in coding regions
  - $FN(X)$ is freq of $X$ in non-coding regions

Dicodon Preference Model Example

- Suppose AAA ATT, AAA GAC, AAA TAG have the following freqs:
  - $FC(\text{AAA ATT}) = 1.4\%$
  - $FN(\text{AAA ATT}) = 5.2\%$
  - $FC(\text{AAA GAC}) = 1.9\%$
  - $FN(\text{AAA GAC}) = 4.8\%$
  - $FC(\text{AAA TAG}) = 0.0\%$
  - $FN(\text{AAA TAG}) = 6.3\%$

  Then
  - $P(\text{AAA ATT}) = -0.57$
  - $P(\text{AAA GAC}) = -0.40$

  Treating STOP codons differently:

  $\Rightarrow$ A region consisting of only these dicodons is probably a non-coding region

Dicodon Preference Model’s Properties

- $P(X) = 0$ if $X$ has same freq in coding and non-coding regions
- $P(X) > 0$ if $X$ has higher freq in coding than in non-coding region; the larger the diff, the more positive the score is
- $P(X) < 0$ if $X$ has higher freq in non-coding than in coding region; the larger the diff, the more negative the score is

Frame-Insensitive Coding Region Preference Model

- A frame-insensitive coding preference $S_M(R)$ of a region $R$ can be defined as

  \[ S_M(R) = \sum_X P(X) \]

  where $X$ is a dicodon in $R$

- $R$ is predicted as coding region if $S_M(R) > 0$

NB. This model is not commonly used
The in-frame + i preference value \( P_i(X) \) of a dicodon \( X \) is defined as:

\[
P_i(X) = \log \frac{FC_i(X)}{FN(X)}
\]

where

- \( FC_i(X) \) is freq of \( X \) in coding regions at in-frame + i positions
- \( FN(X) \) is freq of \( X \) in non-coding regions

R is predicted as coding if

\[
\sum_{i=0,1,2} S_i(R)/|R| > 0
\]

This coding preference model is commonly used.

The in-frame coding region preference model can be defined as:

\[
S_i(R) = \sum X \text{ is a dicodon at in-frame + i position in } R P_i(X)
\]

\( R \) is predicted as coding if

\[
\sum_{i=0,1,2} S_i(R)/|R| > 0
\]

Problem with Coding Region Boundaries

- Making the call: coding or non-coding and where the boundaries are

Need training set with known coding and non-coding regions to select threshold that includes as many known coding regions as possible, and at the same time excludes as many known non-coding regions as possible.

Types of Coding Region Boundaries

- Knowing boundaries of coding regions helps identify them more accurately
- Possible boundaries of an exon

- Splice junctions:
  - Donor site: coding region | GT
  - Acceptor site: CAG | TAG | coding region

- Translation start
  - in-frame ATG

Signals for Coding Region Boundaries

- Splice junction sites and translation starts have certain distribution profiles
- For example, ...

Coding Region Prediction: An Example Procedure

- Calculate all ORFs of a DNA segment
- For each ORF:
  - Slide thru ORF w/ increment of 10bp
  - Calculate in-frame coding region preference score, in same frame as ORF, within window of 60bp
  - Assign score to center of window

E.g., forward strand in a particular frame...

What do you expect at translation stop?
Acceptor Site (Human Genome)

- If we align all known acceptor sites (with their splice junction site aligned), we have the following nucleotide distribution:

<table>
<thead>
<tr>
<th></th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>54.4</td>
<td>60.4</td>
<td>3.2</td>
<td>3.0</td>
<td>0.0</td>
<td>30.6</td>
<td>73.5</td>
<td>7.1</td>
<td>16.0</td>
</tr>
<tr>
<td>C</td>
<td>26.3</td>
<td>12.0</td>
<td>3.3</td>
<td>3.3</td>
<td>0.0</td>
<td>2.8</td>
<td>7.8</td>
<td>33.5</td>
<td>14.5</td>
</tr>
<tr>
<td>G</td>
<td>16.3</td>
<td>12.5</td>
<td>30.2</td>
<td>10.0</td>
<td>0.0</td>
<td>6.1</td>
<td>13.8</td>
<td>31.6</td>
<td>20.9</td>
</tr>
<tr>
<td>T</td>
<td>11.4</td>
<td>24.2</td>
<td>12.5</td>
<td>6.4</td>
<td>0.0</td>
<td>2.5</td>
<td>2.9</td>
<td>5.9</td>
<td>46.2</td>
</tr>
</tbody>
</table>

- Acceptor site: CAG | TAG | coding region

Donor Site (Human Genome)

- If we align all known donor sites (with their splice junction site aligned), we have the following nucleotide distribution:

<table>
<thead>
<tr>
<th></th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>36.7</td>
<td>60.4</td>
<td>7.2</td>
<td>6.7</td>
<td>0.0</td>
<td>50.4</td>
<td>73.5</td>
<td>7.3</td>
<td>16.0</td>
</tr>
<tr>
<td>C</td>
<td>36.1</td>
<td>12.9</td>
<td>3.7</td>
<td>4.3</td>
<td>0.0</td>
<td>2.8</td>
<td>7.8</td>
<td>33.5</td>
<td>14.5</td>
</tr>
<tr>
<td>G</td>
<td>19.1</td>
<td>12.5</td>
<td>30.2</td>
<td>10.0</td>
<td>0.0</td>
<td>6.1</td>
<td>13.8</td>
<td>31.6</td>
<td>20.9</td>
</tr>
<tr>
<td>T</td>
<td>11.4</td>
<td>24.2</td>
<td>12.5</td>
<td>6.4</td>
<td>0.0</td>
<td>2.5</td>
<td>2.9</td>
<td>5.9</td>
<td>46.2</td>
</tr>
</tbody>
</table>

- Donor site: coding region | GT

What Positions Have “High” Info Content?

- For a weight matrix, information content of each column is calculated as:
  \[-\sum_{X \in \{A,C,G,T\}} F(X) \log \left( \frac{F(X)}{0.25} \right)\]

- When a column has evenly distributed nucleotides, its information content is lowest

- Only need to look at positions having high information content

Information Content Around Donor Sites in Human Genome

<table>
<thead>
<tr>
<th></th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>54.1</td>
<td>40.4</td>
<td>3.7</td>
<td>3.2</td>
<td>0.0</td>
<td>30.6</td>
<td>73.5</td>
<td>7.1</td>
<td>16.0</td>
</tr>
<tr>
<td>C</td>
<td>36.1</td>
<td>12.9</td>
<td>3.7</td>
<td>4.3</td>
<td>0.0</td>
<td>2.8</td>
<td>7.8</td>
<td>33.5</td>
<td>14.5</td>
</tr>
<tr>
<td>G</td>
<td>19.1</td>
<td>12.5</td>
<td>30.2</td>
<td>10.0</td>
<td>0.0</td>
<td>6.1</td>
<td>13.8</td>
<td>31.6</td>
<td>20.9</td>
</tr>
<tr>
<td>T</td>
<td>11.4</td>
<td>24.2</td>
<td>12.5</td>
<td>6.4</td>
<td>0.0</td>
<td>2.5</td>
<td>2.9</td>
<td>5.9</td>
<td>46.2</td>
</tr>
</tbody>
</table>

- Information content
  - column -3 = \(-.34 \log (.34/25) - .363 \log (.363/25) - .183 \log (.183/25) - .114 \log (.114/25) = 0.04\)
  - column -1 = \(-.092 \log (.092/25) - .03 \log (.03/25) - .803 \log (.803/25) - .073 \log (.073/25) = 0.30\)

Weight Matrix Model for Splice Sites

- Weight matrix model
  - Build a weight matrix for donor, acceptor, translation start site, respectively
  - Use positions of high information content

<table>
<thead>
<tr>
<th></th>
<th>-3</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>24.3</td>
<td>30.8</td>
<td>18.1</td>
<td>23.0</td>
<td>16.9</td>
<td>30.7</td>
<td>27.6</td>
<td>17.1</td>
<td>30.5</td>
</tr>
<tr>
<td>C</td>
<td>14.5</td>
<td>10.5</td>
<td>20.1</td>
<td>12.7</td>
<td>6.7</td>
<td>3.3</td>
<td>28.7</td>
<td>8.8</td>
<td>2.4</td>
</tr>
<tr>
<td>G</td>
<td>4.5</td>
<td>7.5</td>
<td>18.2</td>
<td>30.2</td>
<td>10.2</td>
<td>3.1</td>
<td>17.7</td>
<td>6.3</td>
<td>18.7</td>
</tr>
<tr>
<td>T</td>
<td>30.1</td>
<td>60.6</td>
<td>30.1</td>
<td>10.0</td>
<td>0.0</td>
<td>30.0</td>
<td>6.7</td>
<td>3.1</td>
<td>30.0</td>
</tr>
</tbody>
</table>

- Extract FIC in the rest of the table

Just to make sure you know what I mean …

- Give me 3 DNA seq of length 10:
  - Seq1 = ACCCGACTCT
  - Seq2 = AGTGTACCTG
  - Seq3 = AGTTCGTATG

- Then the weight matrix is …
Splice Site Prediction: A Procedure

- Add up freq of corr letter in corr positions:
  - AAGGTAAGT: 0.34 + 0.60 + 0.80 + 1.0 + 1.0 + 0.52 + 0.71 + 0.81 + 0.46 = 6.24
  - TGTGTCTCA: 0.11 + 0.12 + 0.03 + 1.0 + 1.0 + 0.02 + 0.07 + 0.05 + 0.16 = 2.56

- Make prediction on splice site based on some threshold

Other Factors Considered by GRAIL

- G+C composition affects codon distributions
- Length of exons follows certain distribution
- Other signals associated with coding regions
  - periodicity
  - structure information
  - pseudo genes

Info Fusion by ANN in GRAIL

Remaining Challenges in GRAIL

- Initial exon
- Final exon
- Indels & frame shifts

Indel & Frame-Shift in Coding Regions

- Problem definition
- Indel & frameshift identification
- Indel correction
- An iterative strategy

Indels in Coding Regions

- Indel = insertion or deletion in coding region
- Indels are usually caused by seq errors
  - ATG GAT CCA CAT ---- → ATG GAT CA CAT ----
  - ATG GAT CTC CA CAT ----
Effects of Indels on Exon Prediction

- Indels may cause shifts in reading frames & affect prediction algs for coding regions

Key Idea for Detecting Frame-Shift

- Preferred reading frame is reading frame w/ highest coding score
- Diff DNA segments may have diff preferred reading frames

⇒ Segment a coding sequence into regions w/ consistent preferred reading frames corr well w/ indel positions
⇒ Indel identification problem can be solved as a sequence segmentation problem!

Frame-Shift Detection by Seq Segmentation

- Partition seq into segs so that
  - Chosen frames of adjacent segs are diff
  - Each segment has >30 bps to avoid small fluctuations
  - Sum of coding scores in the chosen frames over all segments is maximized

Frame-Shift Detection: C(i, r)

- To calculate C(i, r), there are 3 possible cases for each position i:
  - Case 1: no indel occurred at position i
  - Case 2: a base has been deleted in front of ai
  - Case 3: ai is an inserted base
⇒ C(i, r) = max { Case 1, Case 2, Case 3 }

Frame-Shift Detection: Case 1

- No indel occurs at position i. Then
  \[ C(i, r) = C(i-1, r) + \sum_{j=1}^{r} P_j(a_{i-j}...a_i) \]
Frame-Shift Detection: Case 2

- \( a_{i+1} \) is an inserted base. Then

\[
C(i, r) = C(i-2, r') + Pr(a_{i-6}...a_{i-2}a_{i})
\]

\( a_1 \ a_2 \ ... \ a_{i-5} \ a_{i-4} \ a_{i-3} \ a_{i-2} \ a_{i-1} \ a_i \)

di-codon preference

Frame-Shift Detection: Case 3

- A base has been deleted in front of \( a_i \). Then

\[
C(i, r) = C(i-1, r'') + Pr'(a_{i-5}...a_{i-1}C) + Pr(a_{i-4}...a_{i-1}Ca)
\]

\( a_1 \ a_2 \ ... \ a_{i-5} \ a_{i-4} \ a_{i-3} \ a_{i-2} \ a_{i-1} \ a_i \)

di-codon preference

Exercise: why is “C” the best choice for the purpose above?

Frame-Shift Detection: Initiation

- Initial conditions,
  \[
  C(k, r) = -\infty, \quad k < 6
  
  C(6, r) = Pr(a_1 ... a_6)
  \]

- This is a dynamic programming (DP) algorithm; the equations are DP recurrences

Exercise: How to modify the recurrence so that each fragment is at least 30bp?

Frame-Shift Detection: Determining Indel Positions

- Calculation of \( \max_{r \in \{0, 1, 2\}} C(i, r) \) gives an optimal segmentation of a DNA sequence
- Tracing back the transition points—viz. case 2 & case 3—gives the segmentation results

Frame-Shift Detection: Determine Coding Regions

- For given \( H_1 \) and \( H_2 \) (e.g., \( 0.25 \) for noncoding and \( 0.75 \) for coding), partition a DNA seq into segs so that each seg has >30 bases & coding values of each seg are consistently closer to one of \( H_1 \) or \( H_2 \) than the other

\( H_1 \) \( H_2 \)

segmentation result

Frame-Shift Detection: Finally…

- Overlay “preferred reading-frame segs” & “coding segs” gives coding region predictions regions w/ indels
What Happens When Indels Are Close Together?

- Our procedure works well when indels are not too close together (i.e., >30 bases apart)
- When indels are too close together, they will be missed...

Handling Indels That Are Close Together

- Employ an iterative process, viz
  - Find one set of indels
  - Correct them
  - Iterate until no more indels can be found

About the Inventor: Ying Xu

- Regents-GRA Eminent Scholar Chair
  Professor, Dept. of Biochem & Mol Biol, Univ of Georgia, Athens
- Director, Inst of Bioinformatics, Univ of Georgia, Athens

Any Question?

Acknowledgements

- I “borrowed” a lot of materials in this lecture from Xu Ying (Univ of Georgia) and Mark Craven (Univ of Wisconsin)

References

- D. J. States, W. Gish, “Combined use of sequence similarity and codon bias for coding region identification”, JCB, 1:39--50, 1994
References

• V. Solovyev et al. “Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames”, NAR, 22:5156–5163, 1994