Comparative Gene Finding

BMI/CS 776
www.biostat.wisc.edu/bmi776/
Spring 2012
Colin Dewey
cdewey@biostat.wisc.edu
Goals for Lecture

the key concepts to understand are the following:

• using related genomes as an additional source of evidence for gene finding
• the TWINS SCAN approach: use a pre-computed conservation sequence that is aligned to the given DNA sequence
• pair HMMs
• the correspondence between Viterbi in a pair HMM and standard dynamic programming for sequence alignment
• the SLAM approach: use a pair HMM to simultaneously align and parse sequences
Why use comparative methods?

- genes are among the most conserved elements in the genome
  ⇒ use conservation to help infer locations of genes

- some signals associated with genes are short and occur frequently
  ⇒ use conservation to eliminate from consideration false candidate sites
TWINSCAN
Korf et al., Bioinformatics 2001

• prediction with TWINSCAN
given: a sequence to be parsed, $x$
using BLAST, construct a conservation sequence, $c$
have HMM simultaneously parse (using Viterbi) $x$ and $c$

• training with TWINSCAN
given: set of training sequences $X$
for each $x$ in $X$
construct a conservation sequence $c$ for $x$
infer emission parameters for both $x$ and $c$
Conservation Sequences in TWINSCAN

• before processing a given sequence, TWINSCAN first computes a corresponding conservation sequence

\[
\begin{align*}
\text{ATTTAGCCTACTGAAATGGACCGCTTCAGCATGGTATCC} \\
\text{\vdash \vdash \vdash \vdash \vdash \vdash \vdash \vdash \vdash \vdash} \\
\text{matched unaligned mismatched}
\end{align*}
\]

Given: a sequence of length \( n \), a set of aligned BLAST matches \( c[1...n] = \text{unaligned} \)

sort BLAST matches by alignment score

for \( i = 1 \) to \( n \)

for each BLAST match \( h \) (from best to worst)

if \( h \) extends to position \( i \)

if \( c[i] == \text{unaligned} \)

\( c[i] = h[i] \)
Conservation Sequence Example

given sequence

ATTTAGCCTA
ATTTA
AGAAGGGTCACC
resulting conservation sequence

ATTTAGCCTACTGAAATGGACCGCTTCAGCATGGTATCC

significant BLAST matches ordered from best to worst

ATGGACCGCTTCAGC
ACGCACCGCTTCATC
AGCATA
AGAAGGGTCACC
Parsing a DNA Sequence

The Viterbi path represents a parse of a given sequence, predicting exons, introns, etc.
Modeling Sequences in TWINSCAN

• each state “emits” two sequences
  – the given DNA sequence, $x$
  – the conservation sequence, $c$

• it treats them as conditionally independent given the state

$$\Pr(x_i, c_i \mid q) = \Pr(d_i \mid q) \ \Pr(x_i \mid q, d_i) \ \Pr(c_i \mid q, d_i)$$

$$x_i \quad \text{ATTTAGCCTACTGAAATGGACCGCTTCCAGCATGGTATCC}$$

$$c_i \quad \text{..........}$$

$$d_i$$
Modeling Sequences in TWINSCAN

• conservation sequence is treated just as a string over a 3-character alphabet (|, :, .)

• conservation sequence emissions modeled by
  – inhomogeneous 2\textsuperscript{nd}-order chains for splice sites
  – homogeneous 5\textsuperscript{th}-order Markov chains for other states

• like GENSCAN, based on hidden semi-Markov models

• algorithms for learning, inference same as GENSCAN
TWINSCAN vs. GENSCAN

Conservation is neither necessary nor sufficient to predict an exon.

TWINSCAN correctly omits this exon prediction because the conserved region ends within it.

TWINSCAN correctly predicts both splice sites because they are within the aligned regions.
GENSCAN vs. TWINSCAN: Empirical Comparison

sensitivity (Sn) = \( \frac{TP}{TP + FN} \)

specificity (Sp) = \( \frac{TP}{TP + FP} \)

note: the definition of specificity here is somewhat nonstandard; it’s the same as precision

Figure from Flicek et al., *Genome Research*, 2003
Accuracy of TWINSCAN as a Function of Sequence Coverage

The accuracy of TWINSCAN is shown as a function of sequence coverage. The graph compares the sensitivity and specificity for different sequence coverage levels:

- GenScan (none)
- 1X mouse
- 2X mouse
- 3X mouse
- 4X mouse
- Assembly
- Syntenic Regions

The y-axis represents the percentage, ranging from 0% to 16%. The x-axis lists the different sequence coverage levels.

- Very crude mouse genome sequence:
  - Sensitivity: Lower than other coverage levels
  - Specificity: Lower than other coverage levels

- Good mouse genome sequence:
  - Sensitivity: Higher than other coverage levels
  - Specificity: Higher than other coverage levels
prediction with SLAM

given: a pair of sequences to be parsed, \( x \) and \( y \)
find approximate alignment of \( x \) and \( y \)
run constrained Viterbi to have HMM simultaneously parse and align \( x \) and \( y \)

training with SLAM

given: a set of aligned pairs of training sequences \( X \)
for each \( x, \ y \) in \( X \)
\[ \text{infer emission/alignment parameters for both } x \text{ and } y \]
Pair Hidden Markov Models

- each non-silent state emits one or a pair of characters

H: homology (match) state
I: insert state
D: delete state
PHMM Paths = Alignments

sequence 1: AAGCGC
sequence 2: ATGTC

hidden: B H H I I H D H E
observed: A A G C G C A T G T C
Transition Probabilities

- probabilities of moving between states at each step

<table>
<thead>
<tr>
<th></th>
<th>B</th>
<th>H</th>
<th>I</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>B</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>H</strong></td>
<td></td>
<td>1-2δ-τ</td>
<td>δ</td>
<td>δ</td>
<td>τ</td>
</tr>
<tr>
<td><strong>I</strong></td>
<td>1-2δ-τ</td>
<td>δ</td>
<td>δ</td>
<td>τ</td>
<td></td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>1-ε-τ</td>
<td>ε</td>
<td></td>
<td>τ</td>
<td></td>
</tr>
<tr>
<td><strong>E</strong></td>
<td>1-ε-τ</td>
<td>ε</td>
<td></td>
<td>τ</td>
<td></td>
</tr>
</tbody>
</table>
# Emission Probabilities

<table>
<thead>
<tr>
<th>Deletion (D)</th>
<th>Insertion (I)</th>
<th>Homology (H)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$e_D(x_i)$</td>
<td>$e_I(y_j)$</td>
<td>$e_H(x_i, x_j)$</td>
</tr>
<tr>
<td>A</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.2</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>0.2</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>0.1</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>0.4</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>0.1</td>
<td></td>
</tr>
<tr>
<td>A C G T</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.13 0.03 0.06 0.03</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.03 0.13 0.03 0.06</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.06 0.03 0.13 0.03</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.03 0.06 0.03 0.13</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- single character
- single character
- pairs of characters
PHMM Viterbi

- probability of most likely sequence of hidden states generating length $i$ prefix of $x$ and length $j$ prefix of $y$, with the last state being:

$$v^H(i, j) = e_H(x_i, y_j) \max \begin{cases} v^H(i - 1, j - 1)t_{HH}, \\ v^I(i - 1, j - 1)t_{IH}, \\ v^D(i - 1, j - 1)t_{DH} \end{cases}$$

$$v^I(i, j) = e_I(y_j) \max \begin{cases} v^H(i, j - 1)t_{HI}, \\ v^I(i, j - 1)t_{II}, \\ v^D(i, j - 1)t_{DI} \end{cases}$$

$$v^D(i, j) = e_D(x_i) \max \begin{cases} v^H(i - 1, j)t_{HD}, \\ v^I(i - 1, j)t_{ID}, \\ v^D(i - 1, j)t_{DD} \end{cases}$$

- note that the recurrence relations here allow $I \rightarrow D$ and $D \rightarrow I$ transitions
PHMM Alignment

- calculate probability of most likely alignment

\[ v^E(m, n) = \max(v^M(m, n)t_{HE}, v^I(m, n)t_{IE}, v^D(m, n)t_{DE}) \]

- traceback, as in Needleman-Wunsch (NW), to obtain sequence of state states giving highest probability

\[ \text{HIDHHDDIIHH...} \]
Correspondence with NW

- NW values $\approx$ logarithms of PHMM Viterbi values

\[
\begin{align*}
\log v^H(i, j) &= \log e_H(x_i, y_j) + \max \quad \{ \\
&\quad \log v^H(i - 1, j - 1) + \log t_{HH}, \\
&\quad \log v^I(i - 1, j - 1) + \log t_{IH}, \\
&\quad \log v^D(i - 1, j - 1) + \log t_{DH} \\
\end{align*}
\]

\[
\begin{align*}
\log v^I(i, j) &= \log e_I(y_j) + \max \quad \{ \\
&\quad \log v^H(i, j - 1) + \log t_{HI}, \\
&\quad \log v^I(i, j - 1) + \log t_{II}, \\
&\quad \log v^D(i, j - 1) + \log t_{DI} \\
\end{align*}
\]

\[
\begin{align*}
\log v^D(i, j) &= \log e_D(x_i) + \max \quad \{ \\
&\quad \log v^H(i - 1, j) + \log t_{HD}, \\
&\quad \log v^I(i - 1, j) + \log t_{ID}, \\
&\quad \log v^D(i - 1, j) + \log t_{DD} \\
\end{align*}
\]
Posterior Probabilities

- there are similar recurrences for the *Forward* and *Backward* values
- from the *Forward* and *Backward* values, we can calculate the posterior probability of the event that the path passes through a certain state $S$, after generating length $i$ and $j$ prefixes
Uses for Posterior Probabilities

- suboptimal sampling of alignments
- posterior probability of pairs of residues being homologous (aligned to each other)
- posterior probability of a residue being gapped
- training model parameters (EM)
Posterior Probabilities

plot of posterior probability of each alignment column
Parameter Training

• supervised training
  – given: sequences and correct alignments
  – do: calculate parameter values that maximize joint likelihood of sequences and alignments

• unsupervised training
  – given: sequence pairs, but no alignments
  – do: calculate parameter values that maximize marginal likelihood of sequences (sum over all possible alignments)
Generalized Pair HMMs

- represent a parse \( \pi \), as a sequence of states and a sequence of associated lengths for each input sequence

\[
\tilde{q} = \{q_1, q_2, \ldots, q_n\} \\
\tilde{d} = \{d_1, d_2, \ldots, d_n\} \\
\tilde{e} = \{e_1, e_2, \ldots, e_n\}
\]

may be gaps in the sequences
Generalized Pair HMMs

• representing a parse $\pi$, as a sequence of states and associated lengths (durations)

\[ \tilde{q} = \{q_1, q_2, \ldots, q_n\} \]
\[ \tilde{d} = \{d_1, d_2, \ldots, d_n\} \quad \tilde{e} = \{e_1, e_2, \ldots, e_n\} \]

• the joint probability of generating parse $\pi$ and sequences $x$ and $y$

\[
P(x, y, \pi) = a_{start,1}P(d_1, e_1 \mid q_1)P(x_1, y_1 \mid q_1, d_1, e_1) \times \prod_{k=2}^{n} a_{k-1,k}P(d_k, e_k \mid q_k)P(x_k, y_k \mid q_k, d_k, e_k)
\]
Generalized Pair HMM Algorithms

- Generalized HMM Forward Algorithm

\[ f_l(i) = \sum_{k} \sum_{d=1}^{D} \left[ f_k(i-d) \ a_{kl} \ P(d \mid q_l) \ P(x_{i-d+1}^i \mid q_l, d) \right] \]

- Generalized Pair HMM Algorithm

\[ f_l(i, j) = \sum_{k} \sum_{d=1}^{D} \sum_{e=1}^{D} \left[ f_k(i-d, j-e) \ a_{kl} P(d, e \mid q_l) \ P(x_{i-d+1}^i, y_{j-e+1}^j \mid q_l, d, e) \right] \]

- Viterbi: replace sum with max
Prediction in SLAM

- could find alignment and gene predictions by running Viterbi
- to make it more efficient
  - find an approximate alignment (using a fast anchor-based approach)
  - each base in $x$ constrained to align to a window of size $h$ in $y$
- analogous to banded alignment methods
### GENSCAN, TWINSCAN, & SLAM

<table>
<thead>
<tr>
<th>Test set</th>
<th>SN</th>
<th>SP</th>
<th>AC</th>
<th>SN</th>
<th>SP</th>
<th>(SN+SP)/2</th>
<th>ME</th>
<th>WE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>The ROSETTA set</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ROSETTA</td>
<td>0.935</td>
<td>0.978</td>
<td>0.949</td>
<td>0.833</td>
<td>0.829</td>
<td>0.831</td>
<td>0.048</td>
<td>0.047</td>
</tr>
<tr>
<td>SGP-1</td>
<td>0.940</td>
<td>0.960</td>
<td>0.940</td>
<td>0.700</td>
<td>0.760</td>
<td>0.730</td>
<td>0.120</td>
<td>0.040</td>
</tr>
<tr>
<td>SLAM</td>
<td>0.951</td>
<td>0.981</td>
<td>0.960</td>
<td>0.783</td>
<td>0.755</td>
<td>0.769</td>
<td>0.038</td>
<td>0.057</td>
</tr>
<tr>
<td>TWINSCAN,p</td>
<td>0.960</td>
<td>0.941</td>
<td>0.940</td>
<td>0.855</td>
<td>0.824</td>
<td>0.840</td>
<td>0.045</td>
<td>0.081</td>
</tr>
<tr>
<td>TWINSCAN</td>
<td>0.984</td>
<td>0.889</td>
<td>0.923</td>
<td>0.839</td>
<td>0.767</td>
<td>0.803</td>
<td>0.034</td>
<td>0.118</td>
</tr>
<tr>
<td>GENSCAN</td>
<td>0.975</td>
<td>0.908</td>
<td>0.929</td>
<td>0.817</td>
<td>0.770</td>
<td>0.793</td>
<td>0.057</td>
<td>0.107</td>
</tr>
<tr>
<td><strong>HoxA</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLAM</td>
<td>0.852</td>
<td>0.896</td>
<td>0.864</td>
<td>0.727</td>
<td>0.533</td>
<td>0.630</td>
<td>0.000</td>
<td>0.333</td>
</tr>
<tr>
<td>TWINSCAN,p</td>
<td>0.976</td>
<td>0.829</td>
<td>0.896</td>
<td>0.773</td>
<td>0.531</td>
<td>0.652</td>
<td>0.000</td>
<td>0.312</td>
</tr>
<tr>
<td>TWINSCAN</td>
<td>0.949</td>
<td>0.511</td>
<td>0.704</td>
<td>0.591</td>
<td>0.173</td>
<td>0.382</td>
<td>0.000</td>
<td>0.707</td>
</tr>
<tr>
<td>SGP-2</td>
<td>0.640</td>
<td>0.637</td>
<td>0.619</td>
<td>0.409</td>
<td>0.173</td>
<td>0.291</td>
<td>0.091</td>
<td>0.596</td>
</tr>
<tr>
<td>GENSCAN</td>
<td>0.932</td>
<td>0.687</td>
<td>0.796</td>
<td>0.545</td>
<td>0.235</td>
<td>0.390</td>
<td>0.000</td>
<td>0.569</td>
</tr>
<tr>
<td><strong>Elastin</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLAM</td>
<td>0.876</td>
<td>0.981</td>
<td>0.926</td>
<td>0.802</td>
<td>0.859</td>
<td>0.831</td>
<td>0.121</td>
<td>0.059</td>
</tr>
<tr>
<td>TWINSCAN,p</td>
<td>0.942</td>
<td>0.950</td>
<td>0.945</td>
<td>0.879</td>
<td>0.889</td>
<td>0.884</td>
<td>0.066</td>
<td>0.056</td>
</tr>
<tr>
<td>TWINSCAN</td>
<td>0.933</td>
<td>0.877</td>
<td>0.903</td>
<td>0.835</td>
<td>0.826</td>
<td>0.831</td>
<td>0.110</td>
<td>0.120</td>
</tr>
<tr>
<td>SGP-2</td>
<td>0.755</td>
<td>0.998</td>
<td>0.873</td>
<td>0.593</td>
<td>0.900</td>
<td>0.291</td>
<td>0.352</td>
<td>0.017</td>
</tr>
<tr>
<td>GENSCAN</td>
<td>0.947</td>
<td>0.766</td>
<td>0.852</td>
<td>0.835</td>
<td>0.731</td>
<td>0.783</td>
<td>0.121</td>
<td>0.231</td>
</tr>
</tbody>
</table>

The measures of sensitivity SN = TP/TP + FN and specificity SP = TP/TP + FP (where TP = true positives, TN = true negatives, FP = false positives and FN = false negatives) are shown at both the nucleotide and exon level. ME is entirely missed exons, WE is wrong exons, and the approximate correlation AC = 1/2 (TP/TP + FN + TP/TP + FP + TN/TN + FP + TN/TN + FN) – 1 summarizes the overall nucleotide sensitivity and specificity by one number. Within each of the three data sets the methods are divided into three classes: those operating on a syntenic DNA pair, those operating on a human sequence using as evidence matches against a database of mouse sequences, and a single-organism gene finder (GENSCAN).
TWINSCAN vs. SLAM

- both use multiple genomes to predict genes
- both use generalized HMMs
- TWINSCAN
  - takes as an input a genomic sequence, and a conservation sequence computed from an informant genome
  - models probability of both sequences; assumes they're conditionally independent given the state
  - predicts genes only in the genomic sequence
- SLAM
  - takes as input two genomic sequences
  - models joint probability of pairs of aligned sequences
  - can simultaneously predict genes and compute alignments