Eukaryotic Gene Finding: The GENSCAN System

BMI/CS 776
www.biostat.wisc.edu/bmi776/
Spring 2009
Mark Craven
craven@biostat.wisc.edu

The GENSCAN HMM for Eukaryotic Gene Finding [Burge & Karlin ‘97]

Each shape represents a functional unit of a gene or genomic region

Pairs of intron/exon units represent the different ways an intron can interrupt a coding sequence (after 1st base in codon, after 2nd base or after 3rd base)

Complementary submodel (not shown) detects genes on opposite DNA strand

Figure from Burge & Karlin, Journal of Molecular Biology, 1997
The GENSCAN HMM

• for each sequence type, GENSCAN models
  – the length distribution
  – the sequence composition

• length distribution models vary depending on sequence type
  * nonparametric (using histograms)
  – parametric (using geometric distributions)
  – fixed-length

• sequence composition models vary depending on type
  – 5th-order, inhomogeneous
  – 5th-order homogeneous
  – 1st-order inhomogeneous
  * tree-structured variable memory (MDD)

The GENSCAN HMM

• semi-Markov models are well motivated for some sequence elements (e.g. exons)

• dependency structure of splice sites motivates the use of MDD models, which can represent context-specific dependencies
Length Distributions of Introns/Exons

- (a) Introns
- (b) Initial exons
- (c) Internal exons
- (d) Terminal exons

- Geometric distribution provides good fit

Splice Signals

- donor sites
- acceptor sites

Figures from Yi Xing
Motivation for MDD

- How can we model significant dependencies between non-adjacent positions?

\[
\text{pos } i = T \quad \text{pos } j = G \quad \text{pos } j = C \quad \text{pos } j = A
\]

- compute \( \chi^2 \) values using 2×4 table
  - alternative hypothesis: distribution for column \( j \) depends on what is in column \( i \)
  - null hypothesis: distribution for column \( j \) is the same in both cases

Motivation for MDD

- Table shows \( \chi^2 \) values for pairs of positions around donor sites
  - values marked with * show statistically significant dependency

<table>
<thead>
<tr>
<th>( i )</th>
<th>Con</th>
<th>( j )</th>
<th>( -3 )</th>
<th>( -2 )</th>
<th>( -1 )</th>
<th>( +3 )</th>
<th>( +4 )</th>
<th>( +5 )</th>
<th>( +6 )</th>
<th>Sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-3)</td>
<td>c/a</td>
<td>—</td>
<td>61.8*</td>
<td>—</td>
<td>14.9</td>
<td>—</td>
<td>5.8</td>
<td>20.2*</td>
<td>—</td>
<td>11.2</td>
</tr>
<tr>
<td>(-2)</td>
<td>A</td>
<td>1156*</td>
<td>40.5*</td>
<td>—</td>
<td>20.3*</td>
<td>—</td>
<td>20.2*</td>
<td>—</td>
<td>59.7*</td>
<td>42.9*</td>
</tr>
<tr>
<td>(-1)</td>
<td>G</td>
<td>15.4</td>
<td>82.8*</td>
<td>—</td>
<td>13.0</td>
<td>61.5*</td>
<td>—</td>
<td>41.4*</td>
<td>—</td>
<td>96.6*</td>
</tr>
<tr>
<td>(+3)</td>
<td>a/g</td>
<td>8.6</td>
<td>17.5*</td>
<td>13.1</td>
<td>—</td>
<td>19.3*</td>
<td>1.8</td>
<td>0.1</td>
<td>60.5*</td>
<td></td>
</tr>
<tr>
<td>(+4)</td>
<td>A</td>
<td>218*</td>
<td>56.0*</td>
<td>62.1*</td>
<td>64.1*</td>
<td>—</td>
<td>56.8*</td>
<td>—</td>
<td>260.9*</td>
<td></td>
</tr>
<tr>
<td>(+5)</td>
<td>G</td>
<td>11.6</td>
<td>60.1*</td>
<td>41.9*</td>
<td>93.6*</td>
<td>146.6*</td>
<td>—</td>
<td>33.6*</td>
<td>387.3*</td>
<td></td>
</tr>
<tr>
<td>(+6)</td>
<td>t</td>
<td>22.2*</td>
<td>40.7*</td>
<td>103.8*</td>
<td>26.5*</td>
<td>17.8*</td>
<td>32.6*</td>
<td>—</td>
<td>243.6*</td>
<td></td>
</tr>
</tbody>
</table>
The Maximal Dependence Decomposition (MDD) Approach

- induce a tree that represents the dependency structure apparent in the data
- induce partial position weight matrices for each node and leaf of tree

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.1</td>
<td>0.3</td>
<td>0.1</td>
<td>0.2</td>
<td>0.2</td>
<td>0.4</td>
<td>0.3</td>
<td>0.1</td>
</tr>
<tr>
<td>C</td>
<td>0.5</td>
<td>0.2</td>
<td>0.1</td>
<td>0.1</td>
<td>0.6</td>
<td>0.1</td>
<td>0.2</td>
<td>0.7</td>
</tr>
<tr>
<td>G</td>
<td>0.2</td>
<td>0.2</td>
<td>0.6</td>
<td>0.5</td>
<td>0.1</td>
<td>0.2</td>
<td>0.2</td>
<td>0.1</td>
</tr>
<tr>
<td>T</td>
<td>0.2</td>
<td>0.3</td>
<td>0.2</td>
<td>0.2</td>
<td>0.1</td>
<td>0.3</td>
<td>0.3</td>
<td>0.1</td>
</tr>
</tbody>
</table>

- use the tree + weight matrices to calculate the probability of a given sequence

An MDD Learned Tree

Figure from Burge & Karlin, Journal of Molecular Biology, 1997
The MDD Algorithm: Finding the Tree

Given: a set of aligned training sequences $T$
positions $P = \{1, \ldots, k\}$
tree = find_MDD_subtree($T, P$)

find_MDD_subtree($T, P$)
   for each position $i$ in $P$
      determine the consensus base $C_i$
      calculate dependence between $C_i$, other positions
   if stopping criteria not met
      choose the value of $i$ such that $S_i$ is maximal
      make a node with $C_i$ as the test
      $D_i^+ = \text{sequences in } T \text{ with base } C_i \text{ at position } i$
      $D_i^- = \text{other sequences}$
      left subtree = find_MDD_subtree($D_i^+, P - \{i\}$)
      right subtree = find_MDD_subtree($D_i^-, P - \{i\}$)

Stopping Criteria for MDD

1. the $(k-1)^{th}$ level is reached; no further positions to split on
2. no significant dependencies between positions are detected
3. number of sequences in given subset is sufficiently small
Explaining a Sequence with an MDD Tree

- shown are selected position weight matrices for the tree

\[
\begin{array}{cccccccc}
A & C & G & U &  &  &  &  \\
-3 & 0.3 & 0.4 & 0.2 & 0.5 & 0.1 &  &  \\
-2 & 0.4 & 0.3 & 0.1 & 0.1 & 0.1 &  &  \\
-1 & 0.2 & 0.2 & 0.6 & 0.2 & 0.1 &  &  \\
3 & 0.1 & 0.1 & 0.1 & 0.2 &  &  &  \\
4 &  &  &  &  &  &  &  \\
5 &  &  &  &  &  &  &  \\
6 &  &  &  &  &  &  &  \\
\end{array}
\]

- calculated in each node of the tree

\[
P(x_5)
\]

- if \( x_5 \neq G \), use the weight matrix for \( H_5 \) subset

- else

\[
P(x_{-1})
\]

- if \( x_{-1} \neq G \), use the WM for \( G_5H_{-1} \) subset

- else

\[
Pr(x_{-2})
\]

- from \( G_5G_{-1} \) subset
Explaining a Sequence with an MDD Tree

• using model from previous slide

\[ P(\text{AAGGUCAGU}) = 0.3 \times 0.5 \times 0.7 \times 1 \times 1 \times 0.1 \times 0.5 \times 0.7 \times 0.6 \]

A Graphical View of Dependency Structure

• we can represent the dependency structure of a sequence model as a graph
  – nodes represent sequence positions
  – edges represent dependencies in probability distribution

• the dependency structure of a 0th order Markov chain of length 4 (e.g. a motif model inferred by MEME):

\[ \begin{align*}
  &x_1 &x_2 &x_3 &x_4 \\
  A &0.4 &A &0.4 &A \\
  C &0.1 &C &0.1 &C \\
  G &0.2 &G &0.3 &G \\
  T &0.3 &T &0.2 &T \\
\end{align*} \]

• note: this is different than the transition graph
A Graphical View of Dependency Structure

• 1\textsuperscript{st} order model

\[
x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4
\]

• 2\textsuperscript{nd} order model

\[
x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4
\]

• for a fixed-length model, we could consider arbitrary dependencies

\[
x_1 \leftrightarrow x_2 \leftrightarrow x_3 \leftrightarrow x_4
\]

A Graphical View of Dependency Structure

• MDD allows arbitrary dependencies conditioned on values of certain variables

\[
x_3 = \text{G}
\]

\[
x_4 = \text{G}
\]

\[
x_3 = \text{G}
\]

\[
x_4 = \text{G}
\]

\[
x_1 \leftarrow x_2 \leftarrow x_3 \leftarrow x_4
\]
GENSCAN Conclusions

• HMMs readily enable background knowledge to be incorporated into the model
  – state topology
  – length distributions
  – order of Markov chains

• key technical ideas
  – semi-Markov models (previously developed): can represent arbitrary length distributions
  – MDD: can represent context-specific dependencies