**Cartoon: Transcription Control**

Transcription start site (TSS)

Cis-regulatory Module (CRM)
Task

- **Given:**
  - A set of sequences that share a CRM (the **positive** set)
  - A set of sequences that do not (a **negative** set)
  - A set of tentative binding site descriptions

- **Do:**
  - Describe the binding sites and the relationship among them that defines the positive set of sequences
Related Work

- Segal and Sharan (2004)
- Aerts, et. al. (2003)
- Sinha, et. al. (2003)
- Keleşi, et. al. (2004)
- Noto and Craven
Segal and Sharan (2004)

- Discriminative model
- Motifs are pre-initialized (input set of PWMs)
- CRM represented by a probabilistic graphical model
  - Sequence $\rightarrow$ motif $\rightarrow$ window $\rightarrow$ regulated
  - Each event is thresholded with a logit function
- Probabilistically determine which motifs are likely to appear together
Segal and Sharan (2004): PGM

Each level depends on the previous one, and is activated by a weighted sum and a logit function

\[ x = \sum \text{weight}_i \times \text{parent}_i \]

Window contains the CRM?

Regulated
Segal and Sharan (2004): Parameters

Motif occurrences

Window contains the CRM?

Regulated

P(\ g. M_1, 1, g. M_2, 1)
Segal and Sharan (2004): E-Step

Sequence Data

Motif occurrences

Window contains the CRM?

Regulated

\[ P(g. W | g. M_1, g. M_2) \]

\[ P(g. R | g. W) \]
Segal and Sharan (2004): M-Step

Sequence Data


Motif occurrences

Window contains the CRM?

Regulated

?
Segal and Sharan (2004): CRM structure

- Final model consists of:
  - Set of motif models
  - Weights indicating which motifs are relevant to CRMs
  - Weights indicating which combinations of windows are necessary for CRM regulation
- Distance between motifs captured in fixed-length windows
- Distance to TSS captured in weights of specific window (but these are all equal)
Consider a set of motif models (PWMs), \( \{\Theta_1, \Theta_2, \ldots, \Theta_M\} \)

A CRM model is a set of motif models, e.g. \( \{\Theta_A, \Theta_B, \Theta_C, \Theta_D\} \)

An A* search (branch and bound) algorithm guarantees the CRM model that maximizes the score of the sequence data.
Aerts, et. al. (2003): Scoring Function

\[ S_m(s) = \max_{k_1,k_2,\ldots,k_l} p(t) \times b(t) \times \sum_{i=1}^{l} \log S(t_{ki}) \]

\( k_1, k_2 \equiv \) motif occurrences

\( p(t) \equiv \) penalty for missing motifs

\( b(t) \equiv \) legal CRM (e.g. no overlaps, all motifs within window)

\( S \equiv \) motif occurrence score (how well does DNA match PWM?)
Aerts, et. al. (2003): Example

\[ S_m(s) = \max_{k_1, k_2, \ldots, k_l} p(t) \times b(t) \times \sum_{i=1}^{l} \log S(t_{ki}) \]

- Our CRM model is: \{Θ\(_A\), Θ\(_B\), Θ\(_C\), Θ\(_D\)\}
- The motifs occur in a sequence \textbf{s} as:

```
D A C A B D B C
```

- The best (legal) indicator of this CRM model is here, with a score of \(\frac{3}{4} \times 1 \times \text{sum-of-log-of-scores}\)
Aerts, et. al. (2003): A* Search

- Start with an empty set of motif models
- Keep a sorted queue of current solutions
- Create new solutions by removing a solution from the queue, adding a new motif model
  - Overestimate the score so a good solution cannot be missed
  - If adding new motifs cannot improve the score, the solution can be discarded
- This ensures an optimal-scoring CRM
Aerts, et. al. (2003)

- Distance taken into account (motifs appear within a fixed-length window)
- No CRM structure (Each motif weighted equally)
Sinha, et. al. (2003)

- Sequence data generated by a HMM
- A special “history conscious” HMM (hcHMM)
- Choose the model parameters to maximize the probability of the data
Sinha, et. al. (2003): HMM generates DNA

**Diagram:**
- **State:** $w_b$ emits letters from a background state.
- **States $w_i$ and $w_j$:** Emitted from a $k^{th}$ order Markov model.
- **States $w_1$ to $w_M$:** Emitted from a motif PWM $w_1$.

**Equation:**
\[ 1 - \sum_i p_i \]

**Background State:**
- Start here.
- State emits letters from a $k^{th}$ order Markov model.
Sinha, et al. (2003): hcHMM generates DNA

Placeholder: Same as background state, but last non-background motif was i.

background \( w_i \) \( w_j \)

\[
\begin{align*}
&\text{w}_b \\
&\text{b}_{gi} \\
&\text{b}_{gj} \\
&\text{w}_i \\
&\text{w}_j \\
\end{align*}
\]

\[p_i \]

\[p_{i,j} \]
Use Baum-Welch algorithm to find model parameters from sequence data

CRMs have some structure
- CRM size: within a fixed size $L$
- Inter-motif distance: side-by-side relationships within CRM
Zhou and Wong (2004): Hierarchical Mixture Model

Level 1: Probabilistically generated CRMs

Level 2: Probabilistically generated motifs within a CRM
Zhou and Wong (2004): A Generative Model

Within a CRM, with probability $q_k$, generate motif k from its PWM, $\Theta_k$.

With probability $r$, generate a new CRM of fixed-length $l$.

Within a CRM, with probability $q_0$, generate a letter from the background distribution, $\theta_0$ (a 1st-order Markov chain).

With probability $(1-r)$, do not generate a new CRM, but instead generate a letter from the background distribution, $\theta_0$. 
Zhou and Wong (2004): Iterative Parameter Update

\[ \Psi \]
\[ \Theta_k: \text{PWMs for each motif, } k \]
\[ q_k: \text{probability of each motif, } k \]
\[ w_k: \text{width of each motif, } k \]
\[ r: \text{probability of a CRM} \]

\[ \Theta \]
\[ \kappa \]
\[ q \]
\[ w \]
\[ r \]

\[ M: \text{CRM locations} \]
\[ A: \text{Motif locations within CRMs} \]
Zhou and Wong (2004): Parameter Update

\[
P(x_{[1..m]}) = r \times h(m - l + 1, m) \times P(x_{[1..m-l]}) + (1-r) \times P(x_m|x_{m-1}, \theta_0) \times P(x_{[1..m-1]}) \tag{A}
\]

\[
h(i, m) = q_0 \times P(x_m|x_{m-1}, \theta_0) \times h(i, m-1) + q_1 \times P(x_{m-w_1+1, m} | \Theta_1) \times h(i,m-w_1) + q_2 \times P(x_{m-w_2+1, m} | \Theta_2) \times h(i,m-w_2) + \cdots + q_K \times P(x_{m-w_K+1, m} | \Theta_K) \times h(i,m-w_K) \tag{B}
\]
Once these values are calculated, \textit{probabilistically sample} \( M \) (CRM locations) and \( A \) (motif locations within CRMs).

Once \( M \) and \( A \) are established, recalculcate \( \Psi \), and repeat until convergence.

- Output includes relevant motifs (PWMs)
- CRM a weighted collection of motifs
- Motifs within a fixed-size window
Keleş, et. al. (2004)

- Stresses the interaction between TFBSs
- Uses *logic regression* (*LogicMotif*)
- A motif set, $\mathcal{M}$, is given as input
Keleș, et. al. (2004)

- **Y**: outcome variable
  - e.g., Gene expression
  - e.g., 0 for downregulated, 1 for upregulated
- **$S_{n,m}$**: $0 \equiv$ motif $m$ does not occur in sequence $n$
  $1 \equiv m$ occurs in $n$ more than zero times

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>(1 0 1 1 0)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_2$</td>
<td>(0 1 1 1 0)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_3$</td>
<td>(0 1 0 1 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$S_4$</td>
<td>(0 0 1 1 1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Keleş, et. al. (2004): Regression

- Linear Regression
  \[ E[Y \mid S_n] = \beta_0 + \beta_1 S_{n,1} + \ldots + \beta_m S_{n,m} \]

- Logistic Regression
  \[ E \left[ \log \left( \frac{P(Y = 1 \mid S_n)}{1 - P(Y = 1 \mid S_n)} \right) \right] = \beta_0 + \beta_1 S_{n,1} + \ldots + \beta_m S_{n,m} \]

- Logic Regression
  \[ E[Y \mid S_n] = \beta_0 + \beta_1 L_1 + \ldots + \beta_m L_m + \varepsilon \]
Keleș, et. al. (2004): Logic Regression

$L_1$: 

```
B
  or
    and
      A
    and
      C
```

$L_2$: 

```
A
  or
    and
      B
    and
      C
    and
      D
```

(logical complement)

$S_1 = (1\ 0)$
$S_2 = (1\ 0)$
$S_3 = (0\ 1)$
$S_4 = (0\ 0)$
Keleş, et. al. (2004): Logic Regression

- Simulated annealing used to find good trees
- Tree sizes selected with cross-fold validation
- “And” operator allows logical relationships
- “Not” operator allows identification of repressors
- “Or” operator allows
  - Alternate binding sites
  - Variations in binding sites
CRM is a structured hierarchy of motifs, including:
- Distance constraints
- Motif order constraints
- DNA strand constraints

Tree structure
- 2-levels: AND then OR
Noto and Craven: Model Structure Constraints

Transcription start site (TSS)

Must occur on TEMPLATE strand

< 50bp upstream

< 100bp from TSS

Negated Motif
**Noto and Craven: Search Process**

- Start out by calculating the score of unconstrained single-motif trees
- Keep a sorted queue of the best $K$ solutions

\[
P = \frac{1}{4} \quad R = \frac{1}{3}
\]
Noto and Craven: Search Process

Queue of $K$ best solutions

1. Pop first solution off queue
2. Try adding disjuncts
3. Try adding conjuncts
4. Try tightening constraints

Keep current solution if it has the highest score.

Repeat process until the Queue is empty.
Noto and Craven

- Discriminative model
- Distance constraints, strand constraints are represented explicitly (Including the distance to the TSS).
Discussion

- Distance to TSS is not represented
- Distance between motifs is captured within a fixed-length “window” (Individual distances not explicitly represented)
- Tree-structured representations allow for alternate motifs, representations
- Experiments take limited upstream regions only
- Methods are generative, and can over-represent the control region background
- Motifs sets are pre-processed
- Strand constraints are not included
References


