Statistics of transcriptional regulation

Sündüz Keleş

Department of Statistics
Department of Biostatistics and Medical Informatics
University of Wisconsin, Madison

February 18-27, 2008
Part-I Outline

- Transcriptional regulation of gene expression: Transcription factors; binding sites (motifs); enhancers, other components.
- Motif finding problem.
Central dogma

**Gene expression**: A process by which a gene’s coded information is converted into structures present and operating in a cell.

$$DNA \implies mRNA \implies Protein$$

The expression of the genetic information stored in the DNA molecule occurs in two stages:

- **transcription**, during which DNA is transcribed into mRNA;
- **translation**, during which mRNA is translated to produce a protein.

![Diagram of the Central Dogma](image-url)
### Genetic code

<table>
<thead>
<tr>
<th>First letter</th>
<th>Second letter</th>
<th>Third letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td></td>
<td>UU</td>
<td>UU</td>
</tr>
<tr>
<td></td>
<td>UUC</td>
<td>UUC</td>
</tr>
<tr>
<td></td>
<td>UUA</td>
<td>UUA</td>
</tr>
<tr>
<td></td>
<td>UUG</td>
<td>UUG</td>
</tr>
<tr>
<td>C</td>
<td>CU</td>
<td>CU</td>
</tr>
<tr>
<td></td>
<td>CUC</td>
<td>CUC</td>
</tr>
<tr>
<td></td>
<td>CUA</td>
<td>CUA</td>
</tr>
<tr>
<td></td>
<td>CUG</td>
<td>CUG</td>
</tr>
<tr>
<td>A</td>
<td>AU</td>
<td>AU</td>
</tr>
<tr>
<td></td>
<td>AUC</td>
<td>AUC</td>
</tr>
<tr>
<td></td>
<td>AUA</td>
<td>AUA</td>
</tr>
<tr>
<td></td>
<td>AUG</td>
<td>AUG</td>
</tr>
<tr>
<td>G</td>
<td>GU</td>
<td>GU</td>
</tr>
<tr>
<td></td>
<td>GUC</td>
<td>GUC</td>
</tr>
<tr>
<td></td>
<td>GUA</td>
<td>GUA</td>
</tr>
<tr>
<td></td>
<td>GUG</td>
<td>GUG</td>
</tr>
</tbody>
</table>

AUG: start codon.
Each cell contains a complete copy of the organism’s genome (the same hardware!).

Cells are of many different types and states. E.g. skin, blood, and nerve cells, cancerous cells, etc.

What makes the cells different?

Each cell utilizes only a subset of the whole set of genes. Differential gene expression, i.e., when, where, and how much each gene is expressed.

The mechanism that controls gene expression is called the regulation of gene expression.
Stages of regulation of gene expression

- during chromatin modifications (DNA packaging),
- during transcription control,
- splicing,
- transport and translation control.

Transcriptional control: most common way of regulation; occurs during the transcription phase when the DNA is transcribed into RNA.

Basic elements of transcriptional control:
- Transcription factors,
- DNA binding sites (regulatory motifs), enhancers,
- Promoters.
Complexity of eukaryotic transcriptional regulation

Complexity of eukaryotic transcriptional regulation

Promoter region

Transcription start site
Eukaryotic transcriptional regulation: complex interactions

Promoter region

Transcription start site

E7

Rb

E2F
DNA sequence that acts to change the expression of the gene adjacent to it are *cis-acting*.

A *trans-acting* element acts to change the expression of the gene at a distance.

Promoter (or upstream sequence) elements are cis-acting.

Transcription factors themselves are trans-acting.

We will study identification of cis-acting regulatory elements.
Binding sites (regulatory motifs)

DNA binding proteins (transcription factors) bind to DNA in a sequence specific manner. These short DNA sequences (5-25 base pairs) are called binding sites or regulatory motifs.

Examples of GAL4 binding site: Instances from different genes (from Saccharomyces Cerevisiae Promoter Database (SCPD))

>YBR019C TC GGCGGATACCTTCACCG
>YBR020W CGGGCGGACGATTACCCCG
>YLR081W TATCGGAGCGTAGGCGGCGGAC
>YML051W CGGCATCCTACATGCCG
>YOR120W TCGGTTCAGACAGGTCCCG

Gene expression is transcriptionally regulated by transcription factors binding selectively to their specific binding sites.
How do we represent binding sites?

Using the GAL4 example:

- **By consensus sequences:** CGGNNNNNNNNNNNCGG.
  
  **IUPAC nucleic acid symbols:**
  - A = adenine
  - C = cytosine
  - G = guanine
  - T = thymine
  - U = uracil
  - R = G, A (purine)
  - Y = T, C (pyrimidine)
  - K = G, T (keto)
  - M = A, C (amino)
  - S = G, C
  - W = A, T
  - B = G, T, C
  - D = G, A, T
  - H = A, C, T
  - V = G, C, A
  - N = A, G, C, T (any)
- Extract and align the binding sites.
- Count data for each position of the binding site.

\[
\begin{align*}
>YBR019C & \text{ CGGCGGATACCTTCACCCG } \\
>YBR019C & \text{ CGGCGGATACCTTCACCCG } \\
>YBR020W & \text{ CGGGCGACGATTACCCCG } \\
>YLR081W & \text{ CGGAGCGTAGGCGGCGCGG } \\
>YML051W & \text{ CGGCATCCTACATGCGCCG } \\
>YOR120W & \text{ CGGTTTCAGACAGGTCCCG } \\
\end{align*}
\]

- By position weight matrices (PWMs by Stormo, 1982):

<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>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>0.25</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0.25</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>0.25</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.25</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>0.25</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

\[\Rightarrow\] Independent multinomial distribution for each position.
Information content

- **Information content profiles from PWMs**: Information content at position $w$ is defined as

\[
IC(w) \equiv 2 + \sum_{i=1}^{4} p_{iw} \log_2 p_{iw} = 2 - \text{Entropy},
\]

where $i \in \{1 \equiv A, 2 \equiv C, 3 \equiv G, 4 \equiv T\}$
The information content profile of a PWM is a measure of a binding site’s tolerance for substitution: the higher the IC, the lower the tolerance.

- \( IC(w) \) achieves its maximum of 2 when \( p_{iw} = 1 \) for some \( i \).
- \( IC(w) \) achieves its minimum of 0 when \( p_{iw} = 1/4 \) for all \( i \).
Sequence logos provide a graphical representation of a position specific weight matrix. Logos are defined as follows for each position of a motif (i.e., each column of the corresponding PWM).

- Letters representing the four nucleotides \{A, C, G, T\} are stacked on top of each other.
- Letters are sorted according to their frequencies.
- The height of each letter is proportional to its frequency.
- The height of the entire stack is proportional to the information content at that position.
- The vertical scale is in bits, with a maximum of 2 bits.

\[
\text{Height of letter } i \text{ at position } w \propto p_{iw} IC(w);
\]
\[
\text{Total height of stack at position } w \propto IC(w).
\]
Sequence logos

Figure: Lambda cl and cro binding sites sequence logo. From Shaner et al. (1993)
Sequence logos in R?

```r
> pwm
[1,] 0.071429 0.785714 0.285714 0.785714 0.000000 0.000000 0.214286 0
[2,] 0.642857 0.214286 0.714286 0.000000 0.214286 0.785714 0.714286 1
[3,] 0.000000 0.000000 0.000000 0.000000 0.785714 0.214286 0.000000 0
[4,] 0.285714 0.000000 0.000000 0.214286 0.000000 0.000000 0.071429 0

> apply(pwm, 2, sum)
  [1] 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000
  [9] 1.000000 1.000000 1.000000 1.000000

> library(seqLogo)
> seqLogo(pwm)
> seqLogo(pwm, ic.scale = F)
```
Sequence logos in R?

seqLogo(pwm)

seqLogo(pwm, ic.scale = F)
Data collection for the motif finding problem

Through microarray technology:

- **Genomewide gene expression data:** Relative abundance of gene expression in different cell types is measured.


Through comparative genomics: Multiple genome sequences from related species.
Data collection for motif finding problem

- Based on expression or multiple species data, we extract 500-1000bps upstream of the transcription start sites (TSS).
- ChIP-chip data generates specific coordinates of binding which may not be restricted to upstream of the TSS.
Downloading sequence data from UCSC genome browser

- Small number of coordinates.
  - Go to UCSC genome browser (http://genome.ucsc.edu/).
  - Select species and the version of the genome.
  - Define your custom track (e.g., in the form of a bed file: http://genome.ucsc.edu/goldenPath/help/customTrack.html#BED).

```plaintext
track name=dummyTrack description="Chromosome coordinates list"
chr22 20100000 20100100 region1 1 +
chr22 20100011 20100200 region2 1 +
chr22 20100215 20100400 region3 1 -
chr22 20100350 20100500 region4 1 +
chr22 20100700 20100800 region5 1 +
chr22 20100700 20100900 region6 1 -
```

- Return to table browser.
- Select "group: Custom Tracks"; "track: your track".
- Select "output format: sequence".
- Option to display in browser or get a zipped file.
- Hit get output, set sequence retrieval option if additional upstream/downstream sequences are required.
Downloading sequence data from UCSC genome browser

For a large number of coordinates:

- Download the appropriate fasta files from e.g., ftp://hgdownload.cse.ucsc.edu/goldenPath/hg18/chromosomes.

```
drwxr-x---  2 keles keles  4096 Jan  27 14:02 ./
drwxr-x--- 27 keles keles  4096 Feb  15 16:34 ../
-rw-rw-r--  1 keles keles 120240 Sep  18 2003 11.ooc
-rwxrwrxr-x  1 keles keles  217996 Dec  5 2005 blat*
-rw-r-----  1 keles keles  713886 Jan 27 14:01 blatSuite.33.zip
-rwxrwrxr-x  1 keles keles  65856 Dec  5 2005 faToNib*
-rwxrwrxr-x  1 keles keles  67916 Dec  5 2005 faToTwoBit*
-rwxrwrxr-x  1 keles keles  205848 Dec  5 2005 gfClient*
-rwxrwrxr-x  1 keles keles  116072 Dec  5 2005 gfServer*
-rwxrwrxr-x  1 keles keles   66784 Dec  5 2005 nibFrag*
-rwxrwrxr-x  1 keles keles  114808 Dec  5 2005 pslPretty*
-rwxrwrxr-x  1 keles keles   86160 Dec  5 2005 pslReps*
-rwxrwrxr-x  1 keles keles   86564 Dec  5 2005 pslSort*
-rwxrwrxr-x  1 keles keles   65620 Dec  5 2005 twoBitInfo*
-rwxrwrxr-x  1 keles keles   69964 Dec  5 2005 twoBitToFa*
-rw-rw-r--  1 keles keles    9217 Dec  5 2005 version.doc
-rwxrwrxr-x  1 keles keles  230736 Nov  3 2005 webBlat*
```
Downloading sequence data from UCSC genome browser

- Convert the downloaded fasta (*.fa) files to *nib format.
  
  ```
  faToNib chr1.fa chr1.nib
  ```

  ```
  -rw-r----- 1 keles keles 201011368 Feb 16 2006 chr1.fa
  -rw-r----- 1 keles keles 98534989 Jan 27 14:03 chr1.nib
  ```

- Extracting sequence from *nib files.
  
  ```
  hercules01(115)% nibFrag ../1/chr1.nib 11300000 11300010 - out.fa
  ```

  ```
  hercules01(116)% more out.fa
  >../1/chr1.nib:11300000-11300010
  caaggggtttt
  ```

- "nibFrag" command line can be used within R using the system function (can also be embedded into perl).

- Regulatory Sequence Analysis Tools (RSAT) http://rsat.ulb.ac.be/rsat/ also has sequence extraction features. Most useful if you have gene identifiers.
N unaligned sequences $\mathbf{X}_i = (X_{i,1}, \cdots, X_{i,L_i}), i = 1, \cdots, N$, where $L_i$ is the length of the $i$th sequence.

$\Rightarrow$ Start positions $Y_i$ are unknown!
Multinomial mixture models for the sequence data

**One motif per sequence model:** Lawrence and Reilly (1990)

- Sites are distributed independently with
  \[ P_0 = (p_{01}, \cdots, p_{04}) \] for background sites,
  \[ P_w = (p_{w1}, \cdots, p_{w4}) \] for position \( w \) in the motif, \( w \in \{1, \cdots, W\} \).

- Unknown start site (Unobserved random variable): \( Y_{i,l} = 1 \) if binding site starts at position \( l \) in sequence \( i \), \( Y_{i,l} = 0 \) otherwise, \( l \in \{1, \cdots, L_i - 1 + W\} \).

- Only one motif per sequence \( \sum_l Y_{i,l} = 1 \).

- Uniform start site distribution. \( P(Y_{i,l} = 1) = 1/(L_i - W + 1) \).

**Extension:** Zero or one motif per sequence model: Bailey and Elkan’s MEME (1994).

- Introducing another hidden variable: \( Z_i = 1 \) if sequence \( i \) has one copy of the motif, \( Z_i = 0 \) otherwise.

**Motif finding problem casted as a missing data problem.**
Maximum likelihood estimation with the full (complete) data

Let $\mathcal{P}$ denote the entire parameter set. The log-likelihood function of the unobserved, complete data is

$$l_C \equiv \log Pr(X_1, \cdots X_N, Y_1, \cdots, Y_N; \mathcal{P}) = \sum_{i=1}^{N} \log Pr(X_i, Y_i; \mathcal{P})$$

$$= \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \log Pr(X_i \mid Y_{i,l} = 1; \mathcal{P}) Pr(Y_{i,l} = 1 \mid \mathcal{P}),$$

where background and foreground sequences contribute as

$$Pr(X_i \mid Y_{i,l} = 1, \mathcal{P}) = \prod_{k \in T_i^l} \prod_{j=1}^{4} p_{0j}^{I(X_{i,k}=j)} \prod_{w=1}^{W} \prod_{j=1}^{4} p_{wj}^{I(X_{i,l+w-1}=j)},$$

and $T_i^l \equiv \{1, \cdots, L_i\} \setminus \{l, l+1, \cdots, l+W-1\}$ denotes background sites for a motif start site at position $l$ in sequence $i$, i.e., $Y_{i,l} = 1.$
Maximum likelihood estimation with the full (complete) data

\[ I_C = \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \sum_{k \in T_i^l} \sum_{j=1}^{4} I(X_{i,k} = j) \log(p_{0j}) \]

\[ + \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \sum_{w=1}^{W} \sum_{j=1}^{4} I(X_{i,l+w-1} = j) \log(p_{wj}) \]

Maximize w.r.t. \( p_{0j}, p_{wj}, w = 1, \ldots, W, j = 1, \ldots, 4. \)

For fixed \( j, w \):

\[ \mathcal{L}(p_{wj}, \lambda) = \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) I(X_{i,l+w-1} = j) \log(p_{wj}) \]

\[ + \lambda(1 - \sum_{j=1}^{4} p_{wj}). \]
Maximum likelihood estimation with the full (complete) data

\[
\frac{\partial}{\partial p_{wj}} \mathcal{L}(p_{wj}, \lambda) = \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) I(X_{i,l+w-1} = j) \frac{1}{p_{wj}} - \lambda = 0, \\
\forall j = 1, \ldots, 4.
\]

\[
\hat{p}_{wj} = \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) I(X_{i,l+w-1} = j)}{\sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \sum_{j=1}^{4} I(X_{i,l+w-1} = j)} = \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) I(X_{i,l+w-1} = j)}{N}
\]

(# of times we observe nucleotide \(j\) at the \(w\)-th position of the binding site)/(# of sequences) \(\implies\) Multinomial MLE.
Maximum likelihood estimation with the full (complete) data

\[
\hat{p}_{0j} = \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} I(Y_{i,l} = 1) \sum_{k \in T_i} I(X_{i,k} = j)}{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} I(Y_{i,l} = 1) \sum_{k \in T_i} \sum_{j=1}^{4} I(X_{i,k} = j) / \sum_{i=1}^{N} (L_i - W + 1)}
\]

(\# of times we observe nucleotide \(j\) in a background position/\(# of background positions\) \(\Rightarrow\) Multinominal MLE.)
Incomplete data

With the incomplete observed data, we are lacking $I(Y_{i,l} = 1)$.

$$\hat{p}_{0j} = \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \sum_{k \in T_i} I(X_{i,k} = j)}{\sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \sum_{k \in T_i} \sum_{j=1}^{4} I(X_{i,k} = j)}$$

$$= \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} I(Y_{i,l} = 1) \sum_{k \in T_i} I(X_{i,k} = j)}{\sum_{i=1}^{N} (L_i - W + 1)}$$
Maximum likelihood for the incomplete data

The log-likelihood function of the observed, incomplete data (where $Y_i$ are missing) is

$$l_I \equiv \log \Pr(X_1, \ldots, X_N; \mathcal{P}) = \sum_{i=1}^{N} \log \Pr(X_i; \mathcal{P})$$

$$= \sum_{i=1}^{N} \log \left( \sum_{l=1}^{L_i-W+1} \Pr(X_i \mid Y_{i,l} = 1; \mathcal{P}) \Pr(Y_{i,l} = 1 \mid \mathcal{P}) \right),$$

where $\Pr(Y_{i,l} = 1; \mathcal{P}) = 1/(L_i - W + 1)$.

$\implies$ Direct optimization poses some difficulty. Log-likelihood does not factorize nicely.
Expectation-Maximization (EM) algorithm


- A general method for finding MLEs in a probabilistic model with unobserved (latent) variables.
- Iterates between an Expectation step (E-step) and a Maximization (M-step).
- E-step computes the expected complete data likelihood (i.e., replaces \( I(Y_i, l = 1) \) with their conditional expectations given the observed data and an initial set of parameters) and M-step step maximizes this quantity w.r.t. model parameters.
- Guaranteed to converge to a local optima.
EM Algorithm

- **X**: unobserved complete data with density \( f(x; \Psi) \).
- **Y**: Observed data with density \( g(y; \Psi) = \int_{X(y)} f(x; \Psi) \, dx \).
- Conditional density of \( X \) given \( Y \): \( k(x \mid y; \Psi) = f(x; \Psi)/g(y; \Psi) \).
- Log-likelihood of \( X \): \( l_c(\Psi) \equiv \log(f(x; \Psi)) \).
- Log-likelihood of \( Y \): \( l_l(\Psi) \equiv \log(g(y; \Psi)) \).
- Main function for the EM-algorithm:

\[
Q(\Psi' \mid \Psi) \equiv E[\log f(x; \Psi') \mid y; \Psi] = E[l_c(\Psi') \mid y; \Psi].
\]

- Starting with an initial estimate \( \Psi_0 \) of \( \Psi \), the EM iteration \( \Psi^{\text{old}} \rightarrow \Psi^{\text{new}} \) is as follows:
  - **E-step.** Compute the Q function \( Q(\Psi \mid \Psi^{\text{old}}) \).
  - **M-step.** Select \( \Psi^{\text{new}} \) to maximize the Q-function.

\[
\Psi^{\text{new}} = \arg\max_{\Psi} Q(\Psi \mid \Psi^{\text{old}}).
\]
Main property. The log-likelihood for the observed data
\( l_I(\Psi) = \log g(y; \Psi) \) is non-decreasing at each iteration of the EM
algorithm.
Define
\[
H(\Psi' \mid \Psi) = E[\log k(x \mid y; \Psi') \mid y; \Psi].
\]
Then,
\[
Q(\Psi' \mid \Psi) = l_I(\Psi') + H(\Psi' \mid \Psi).
\]
Proof relies on Jensen’s inequality: For a random variable \( Z \) and concave
function \( h \) (e.g. \( \log \))
\[
E[h(Z)] \leq h(E[Z]).
\]
EM Algorithm: Why does it work?

**Lemma**

For any pair \((\Psi, \Psi')\), \(H(\Psi' \mid \Psi) \leq H(\Psi \mid \Psi)\), with equality iff \(k(x \mid y; \Psi') = k(x \mid y; \Psi)\) a.e. in \(\mathcal{X}\).

**Proof.**

\[
H(\Psi' \mid \Psi) - H(\Psi \mid \Psi) = \mathbb{E}[\log k(x \mid y; \Psi') \mid y; \Psi] - \mathbb{E}[\log k(x \mid y; \Psi) \mid y; \Psi] = \mathbb{E} \left[ \log \frac{k(x \mid y; \Psi')}{k(x \mid y; \Psi)} \mid y; \Psi \right] \\
\leq \log \left\{ \mathbb{E} \left[ \frac{k(x \mid y; \Psi')}{k(x \mid y; \Psi)} \mid y; \Psi \right] \right\} \\
= \log \left\{ \int_{\mathcal{X}(y)} \frac{k(x \mid y; \Psi')}{k(x \mid y; \Psi)} k(x \mid y; \Psi) d(x) \right\} \\
= \log 1 = 0
\]
EM Algorithm: Why does it work?

**Theorem**

The log-likelihood of the observed data is non-decreasing at every EM iteration, i.e.,

\[ l_I(\psi^{\text{new}}) \geq l_I(\psi^{\text{old}}). \]

**Proof.**

Since \( Q(\psi' \mid \psi) = l_I(\psi') + H(\psi' \mid \psi) \) for any pair \((\psi, \psi')\),

\[
\begin{align*}
l_I(\psi^{\text{new}}) - l_I(\psi^{\text{old}}) &= \left\{ Q(\psi^{\text{new}} \mid \psi^{\text{old}}) - Q(\psi^{\text{old}} \mid \psi^{\text{old}}) \right\} \\
&\quad - \left\{ H(\psi^{\text{new}} \mid \psi^{\text{old}}) - H(\psi^{\text{old}} \mid \psi^{\text{old}}) \right\}
\end{align*}
\]

By def. difference in \( Q \) function \( \geq 0 \).

By the lemma difference in \( H \) function \( \leq 0 \).
EM algorithm: notes

- EM is particularly useful when maximum likelihood estimation of a complete data model is easy.
- M-step does not need to find the maximizer of the $Q$-function. EM will still converge if we make sure that $\Psi_{\text{new}}$ provides a higher $Q$-function than that of $\Psi_{\text{old}}$ (Generalized EM).
- Convergence of the EM can be slow. The performance is typically quite sensitive to the initial starting values and the actual amount of missing data.
- EM is a coordinate ascent algorithm. Two coordinates are "unobserved part of $X$" and $\Psi$. E-step takes one step uphill on the "unobserved part of $X$" and M-step on the $\Psi$. In other words, both E and M-steps can be considered as maximization steps.
The main EM Q-function is

\[
Q(\mathcal{P} \mid \mathcal{P}^{\text{old}}) \equiv E[l_{C}(\mathcal{P}) \mid \mathbf{X}_1, \cdots, \mathbf{X}_N; \mathcal{P}^{\text{old}}] \\
= E[\log Pr(X_1, \cdots, X_N, Y_1, \cdots, Y_n; \mathcal{P}) \mid X_1, \cdots, X_N; \mathcal{P}^{\text{old}}] \\
\propto \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} E[I(Y_{i,l} = 1) \mid X_i; \mathcal{P}^{\text{old}}] \log Pr(X_i \mid Y_{i,l} = 1; \mathcal{P}) \\
= \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} E[I(Y_{i,l} = 1) \mid X_i; \mathcal{P}^{\text{old}}] \\
\left\{ \sum_{k \in T_i} \sum_{j=1}^{4} I(X_{ik} = j) \log p_{0j} \\
\sum_{w=1}^{W} \sum_{j=1}^{4} I(X_{i,l+w-1} = j) \log p_{wj} \right\}.
\]
EM algorithm for OOPS

The EM Q-function can be rewritten as

$$Q(\mathcal{P} \mid \mathcal{P}^{\text{old}}) = \sum_{j=1}^{N} N_{0j} \log p_{0j} + \sum_{j=1}^{W} \sum_{w=1}^{W} N_{wj} \log p_{wj},$$

where, for $j = 1, \cdots, 4$,

$$N_{wj} = \begin{cases} \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} E[I(Y_i,l = 1) \mid X_i, \mathcal{P}^{\text{old}}] I(X_i,l+w-1 = j), & w \neq 0 \\ \sum_{i=1}^{N} \sum_{l=1}^{L_i-W+1} E[I(Y_i,l = 1) \mid X_i, \mathcal{P}^{\text{old}}] \sum_{k \in T_l} I(X_i,k = j), & w = 0 \end{cases}$$

The E-step yields the $E[I(Y_i,l = 1) \mid X_i, \mathcal{P}^{\text{old}}]$ required to compute $N_{wj}$. The M-step seeks $\mathcal{P}$, i.e., $p_w, w = 1, \cdots, W$, that maximizes $Q(\mathcal{P} \mid \mathcal{P}^{\text{old}})$ based on the $N_{wj}$ from the E-step.
EM algorithm for OOPS

E-step: OOPS

\[
E[I(Y_{i,l} = 1) \mid X_i, \mathcal{P}^r] = \frac{Pr(X_i \mid Y_{i,l} = 1, \mathcal{P}^{old})}{\sum_{l' = 1}^{L_i - W + 1} Pr(X_i \mid Y_{il'} = 1, \mathcal{P}^{old})}.
\]

M-step: OOPS

\[
p_{0j}^{new} = \frac{N_{0j}}{\sum_{j' = 1}^{4} N_{0j'}} , \quad j \in \{1, \cdots, J\}
\]

\[
= \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} Pr(Y_{i,l} = 1 \mid X_i, \mathcal{P}^r) \sum_{k \in T_l} I(X_i, k = j)}{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} Pr(Y_{i,l} = 1 \mid X_i, \mathcal{P}^r)(L_i - W + 1)},
\]

\[
p_{wj}^{new} = \frac{N_{wj}}{\sum_{j' = 1}^{4} N_{wj'}} , \quad w \in \{1, \cdots, W\}, \quad j \in \{1, \cdots, J\}
\]

\[
= \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} Pr(Y_{i,l} = 1 \mid X_i, \mathcal{P}^r) I(X_i, l + w - 1 = j)}{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} Pr(Y_{i,l} = 1 \mid X_i, \mathcal{P}^r)}
\]

\[
= \frac{\sum_{i=1}^{N} \sum_{l=1}^{L_i - W + 1} Pr(Y_{i,l} = 1 \mid X_i, \mathcal{P}^r) I(X_i, l + w - 1 = j)}{N}.
\]
Output of motif finding

- Estimated position weight matrix (M-step).
- Most likely start sites in each sequence (by-product of the E-step).
- Estimated background distribution (M-step).
Other issues

- Starting values?
- How to choose the motif width $W$?
- More complicated background models? Higher order Markov chain model?
- Extension to any number of occurrences in each sequence?
- Incorporation of "biological insights/known information about the TF family into the model".
Motif finding in action

Data: 16 mouse genomic regions bound by GATA2.

```plaintext
system("more /scratch/Bresnick_Ebox/SeqA_280108/EOBX_WGATAR_BOUND_F150.fasta")

Seq1.Chr6 6:88168826+88168845
cctccgcttatattaaacacacagcgccaccaaaaaacctgcctttttatttttttcatggagtcacctatctg
ttgttatctttgtgatgtttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt...```
library(cosmo)

r1 = cosmo(seqs = "/scratch/Bresnick_Ebox/SeqA_280108/EBOX_WGATAR_BOUND_F150.fasta",
constraints = "None", minW = 5, maxW = 25, models="ZOOPS", revComp =
TRUE)

cvOrder: Order of background Markov model estimated as order = 2 by CV

eGetStart: Extracting starting values from sequence 16/16 fit: mType
  = ZOOPS conSet = 0 width = 25 nSitesNum = 4/4 starting value = 5/5
finalModel: fitting model for width 20 modType ZOOPS and conSet 0

finalModel: startNum 0 and nSitesNum 3 fit: mType = ZOOPS conSet = 0
width = 20 nSitesNum = 4/4 starting value = 1/5
### Output from cosmo

```r
> r1
   1  2  3  4  5  6  7  8  9 10 11 12 13 14
A 0  1 0.1250 0.3749 0 0 0.0000 0.250 0.1875 0.5625 0.3750 0.1875 0.1250 0.375
C 1  0 0.0625 0.5001 0 0 0.1875 0.250 0.3124 0.1250 0.2500 0.1250 0.2500 0.375
G 0  0 0.3750 0.1250 0 1 0.2500 0.125 0.1875 0.2500 0.3125 0.4374 0.4999 0.125
T 0  0 0.4375 0.0000 1 0 0.5625 0.375 0.3126 0.0625 0.0625 0.2500 0.1250 0.125
     15 16 17 18 19 20
A 0.5625 0 1 0 1 0.8750
C 0.0000 0 0 0 0 0.0000
G 0.0000 1 0 0 0 0.0625
T 0.4375 0 0 1 0 0.0625

> slotNames(r1)
[1] "seqs"   "pwm"   "back"   "tmat"   "cand"
[6] "cons"   "sel"   "motifs" "probs" "objectCall"
```
## Accessing the most likely motif occurrences

```r
> r1@motifs

<table>
<thead>
<tr>
<th>seq</th>
<th>pos</th>
<th>orient</th>
<th>motif</th>
<th>prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seq3.Chr6</td>
<td>151</td>
<td>1</td>
<td>CATGTGTACGCTGAAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq20.Chr6</td>
<td>151</td>
<td>1</td>
<td>CAAATGTAAGAGAGTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq28.Chr6</td>
<td>151</td>
<td>1</td>
<td>CACATGGGGGAATGCTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq38.Chr6</td>
<td>151</td>
<td>1</td>
<td>CATATGTGCTGGGCTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq47.Chr6</td>
<td>151</td>
<td>1</td>
<td>CATATGGCCAAGCAAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq4.Chr7</td>
<td>151</td>
<td>1</td>
<td>CAGCTGCTTCACCAGAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq7.Chr7</td>
<td>151</td>
<td>1</td>
<td>CAGATGTCTAATTTCAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq21.Chr7</td>
<td>151</td>
<td>1</td>
<td>CATCTGGACCCGCTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq5.Chr1</td>
<td>151</td>
<td>1</td>
<td>CATCTGGCAGGAACCTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq9.Chr1</td>
<td>151</td>
<td>1</td>
<td>CAGATGTTAATGGAAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq21.Chr1</td>
<td>151</td>
<td>1</td>
<td>CATCTGGTCAGGCAAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq27.Chr1</td>
<td>151</td>
<td>1</td>
<td>CAGCTGTGGGCGTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq33.Chr1</td>
<td>151</td>
<td>1</td>
<td>CAGGTGTCTAAACGGACAGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq1.Chr6</td>
<td>151</td>
<td>1</td>
<td>CATCTGCAGCCGCTGATAA</td>
<td>1</td>
</tr>
<tr>
<td>Seq14.Chr6</td>
<td>137</td>
<td>-1</td>
<td>ATATCTTGAAGCAGCTTTCAGTTG</td>
<td>1</td>
</tr>
<tr>
<td>Seq7.Chr6</td>
<td>151</td>
<td>1</td>
<td>CAGCTGTTTACATTTGAGATAG</td>
<td>1</td>
</tr>
</tbody>
</table>
```
Extension to two-component mixture model (TCM)

- Allows each sequence to contain an arbitrary number of non-overlapping occurrences of the motif.
- Basic idea: A given sequence $X_i$ is generated by repeatedly deciding whether to insert a background nucleotide or a motif width of $W$ (with probability $\lambda$).
- A motif is inserted in either of the one of the two possible orientations with equal probability.
- The likelihood function for the TCM model is a sum over all possible sample paths that could have produced the sequence at hand (in the order of $2^L$).
- Due to this increased computational complexity, exact methods based on the TCM have been avoided.
Main idea is by Bailey and Elkan (1994). Creators of the MEME software.

Derive a new dataset from the original dataset by including all overlapping sequences of length $W$ in the original dataset.

A proportion $\lambda'$ of these derived sequences $X'_i$ represents motifs.

Estimate the parameters of this model based on a modified EM-algorithm that includes a smoothing step after the E-step to reduce the degree to which any two overlapping subsequences can both be assigned to the motif component.

These estimates are then taken as estimates of the parameters in the original model, except that $\lambda$ is estimated by

$$\hat{\lambda} = \frac{1}{\hat{\lambda}' - W + 1}$$

Drawbacks of this approach?

Exact likelihood calculations are possible via a dynamic programming method based on the observed data likelihood (Bembom et al., 2007).
Approximations to TCM

- MEME’s pseudo dataset approach.
- Modification of Keleș et al. (2003) by cosmo (Bembom et al. (2007)) approximation.
Simulation study

- Target motif data consist of 51 human transcription factor position weight matrices from the curated JASPAR database (Sandelin et al., 2004; http://jaspar.genereg.net/).
- Widths range from 5 to 20, with a median of 10 and the average information content profile ranges from 0.76 to 1.73 with a median of 1.21.
- For each motif, 6 data generating models were considered (OOPS, ZOOPS (2), TCM (3)) as data generating models.
- For each motif-model combination 5 datasets, each with 25 750bps sequences were generated.
- Background nucleotides of all the sequences are simulated according to a third order Markov model with transition matrix estimated from the human sequences provided by Tompa et al. (2005).
- For each model setting (6 total), performance is reported over 255 (51 × 5) datasets.
Measuring performance

Sensitivity = $\frac{TP}{TP+FN}$
Specificity = $\frac{TN}{FP+TN}$
Positive Predictive Value = $\frac{TP}{TP+FP}$

ROC curve: Sensitivity vs. 1-Specificity
AUROC: Area under the ROC curve.

For all eligible sites, the estimated PWM is used to compute posterior probabilities. ROC curve is obtained by thresholding these posterior probabilities.
## Table: Mean performance statistics for different approaches to evaluating the TCM likelihood function.

<table>
<thead>
<tr>
<th></th>
<th>Sens</th>
<th>PPV</th>
<th>ROC</th>
<th></th>
<th>Sens</th>
<th>PPV</th>
<th>ROC</th>
<th></th>
<th>Sens</th>
<th>PPV</th>
<th>ROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td>0.14</td>
<td>0.13</td>
<td>0.89</td>
<td>dTCM1</td>
<td>0.29</td>
<td>0.33</td>
<td>0.90</td>
<td>dTCM2</td>
<td>0.42</td>
<td>0.68</td>
<td>0.93</td>
</tr>
<tr>
<td>Keleș</td>
<td>0.31</td>
<td>0.21</td>
<td>0.95</td>
<td></td>
<td>0.50</td>
<td>0.44</td>
<td>0.97</td>
<td></td>
<td>0.54</td>
<td>0.78</td>
<td>0.99</td>
</tr>
<tr>
<td>Bembom</td>
<td>0.33</td>
<td>0.22</td>
<td>0.96</td>
<td></td>
<td>0.51</td>
<td>0.44</td>
<td>0.97</td>
<td></td>
<td>0.54</td>
<td>0.79</td>
<td>0.98</td>
</tr>
<tr>
<td>Exact</td>
<td>0.34</td>
<td>0.22</td>
<td>0.95</td>
<td></td>
<td>0.52</td>
<td>0.44</td>
<td>0.97</td>
<td></td>
<td>0.55</td>
<td>0.78</td>
<td>0.99</td>
</tr>
</tbody>
</table>
Model selection problems involved in motif finding

Which model to fit OOPS, ZOOPS, TCM? How to set $W$?

$p$: #of parameters in the model; $N$: sample size.

- **AIC**: $-2 \log(\text{likelihood of the observed data}) + 2p$
- **BIC**: $-2 \log(\text{likelihood of the observed data}) + p \log(N)$.

**Likelihood-based cross-validation**: with loss function $-\log(\text{likelihood of the observed data})$.

**E-value method.** Measure of the statistical significance of the multiple alignment obtained by aligning the predicted motifs.
Formalizing the model selection problem

Recall that the sequence data $X_1, \cdots, X_n$ are assumed to be $n$ i.i.d. observations from the random variable $X$ with distribution $P$. Let $f$ denote the corresponding density. Let $P_n$ be the empirical distribution function based on $X_1, \cdots, X_n$. Let $\mathcal{M}_k, k = \{1, \cdots, K\}$ be a set of binding site models and let

$$f_k(. \mid P_n) = \arg \max_{f \in \mathcal{M}_k} \int \log(f(x)) dP_n(x),$$

be the maximum likelihood estimator of $f$ according to the model $\mathcal{M}_k$.

This is the result from the EM algorithm for a given $\mathcal{M}_k$. **Goal:** Choose a $\hat{k}$ such that $f_{\hat{k}}(. \mid P_n)$ converges to the true density optimally.
The Kullback-Leibler divergence between densities $f$ and $g$ is defined as

$$KL(f, g) = \int \log \left( \frac{f(x)}{g(x)} \right) dP(x) = \int \log(f(x)) dP(x) - \int \log(g(x)) dP(x).$$

Note that $KL(f, g) \geq 0$ and $KL(f, g) = 0$ iff $f = g$.

Define $\theta_{opt} = -\int \log(f(x)) dP(x)$.

Given $P_n$, goal is to choose $f_k(\cdot | P_n)$ closest to the true $f$.

We would like to choose

$$\tilde{k}_n = \arg\min_{k \in \{1, \ldots, K\}} KL(f, f_k) = \arg\min_{k \in \{1, \ldots, K\}} -\int \log(f_k(x | P_n)) dP(x).$$

$\implies$ $P$ is unknown! Replace $P$ by $P_n$? Overfitting!
Cross-validation method

Define random vector $S_n \in \{0, 1\}^n$ for splitting the sample into a validation and a training sample.

$$S_{n,i} = \begin{cases} 
0 & \text{if i-th observation is in the training sample} \\
1 & \text{if i-th observation is in the validation sample}
\end{cases}$$

Different distributions of $S_n$ cover many types of cross-validation including $V-$ fold cross-validation, Monte Carlo cross-validation.

E.g. 5-fold cross-validation: $S_n$ has 5 realizations.

Let $P_{n,S_n}^0, P_{n,S_n}^1$ denote the empirical distribution of the training and validation sample.

$$p \equiv p(n) = 1/n \sum_{i=1}^n S_{n,i}$$

denote the proportion of the validation sample.
Cross-validated likelihood criteria is given by

\[ \hat{\theta}(k) = -E_{S_n} \int \log(f_k(x \mid P_{n,S_n}^0)) dP_{n,S_n}^1(x) \]

\[ = -E_{S_n} \left[ \frac{1}{np} \sum_{i:S_n,i=1} \log f_k(x_i \mid P_{n,S_n}^0) \right]. \]

and defines an optimal choice \( \hat{k} \)

\[ \hat{k} = \arg\min_{k \in \{1, \ldots, K\}} \hat{\theta}(k). \]
Simulation study: selecting the motif width

Table: Mean performance statistics for different approaches to selecting the motif width.

<table>
<thead>
<tr>
<th></th>
<th>dOOPS</th>
<th></th>
<th>dZOOPS1</th>
<th></th>
<th>dZOOPS2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sens</td>
<td>PPV</td>
<td>ROC</td>
<td>Sens</td>
<td>PPV</td>
</tr>
<tr>
<td>MEME</td>
<td>0.35</td>
<td>0.35</td>
<td>0.92</td>
<td>0.08</td>
<td>0.08</td>
</tr>
<tr>
<td>known</td>
<td>0.57</td>
<td>0.57</td>
<td>0.97</td>
<td>0.23</td>
<td>0.16</td>
</tr>
<tr>
<td>AIC</td>
<td>0.59</td>
<td>0.59</td>
<td>0.97</td>
<td>0.23</td>
<td>0.15</td>
</tr>
<tr>
<td>BIC</td>
<td>0.60</td>
<td>0.60</td>
<td>0.97</td>
<td>0.25</td>
<td>0.17</td>
</tr>
<tr>
<td>likCV</td>
<td>0.54</td>
<td>0.54</td>
<td>0.97</td>
<td>0.26</td>
<td>0.21</td>
</tr>
</tbody>
</table>
Simulation study: selecting the motif width

Table: Mean performance statistics for different approaches to selecting the motif width.

<table>
<thead>
<tr>
<th></th>
<th>Sens</th>
<th>PPV</th>
<th>ROC</th>
<th></th>
<th>Sens</th>
<th>PPV</th>
<th>ROC</th>
<th></th>
<th>Sens</th>
<th>PPV</th>
<th>ROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td>0.13</td>
<td>0.12</td>
<td>0.89</td>
<td>dTCM1</td>
<td>0.28</td>
<td>0.32</td>
<td>0.91</td>
<td>dTCM2</td>
<td>0.41</td>
<td>0.67</td>
<td>0.93</td>
</tr>
<tr>
<td>known</td>
<td>0.33</td>
<td>0.22</td>
<td>0.96</td>
<td></td>
<td>0.51</td>
<td>0.44</td>
<td>0.97</td>
<td></td>
<td>0.54</td>
<td>0.79</td>
<td>0.98</td>
</tr>
<tr>
<td>AIC</td>
<td>0.34</td>
<td>0.21</td>
<td>0.95</td>
<td></td>
<td>0.53</td>
<td>0.46</td>
<td>0.97</td>
<td></td>
<td>0.55</td>
<td>0.79</td>
<td>0.98</td>
</tr>
<tr>
<td>BIC</td>
<td>0.36</td>
<td>0.23</td>
<td>0.96</td>
<td></td>
<td>0.54</td>
<td>0.47</td>
<td>0.97</td>
<td></td>
<td>0.54</td>
<td>0.77</td>
<td>0.98</td>
</tr>
<tr>
<td>likCV</td>
<td>0.34</td>
<td>0.27</td>
<td>0.96</td>
<td></td>
<td>0.51</td>
<td>0.50</td>
<td>0.97</td>
<td></td>
<td>0.54</td>
<td>0.78</td>
<td>0.98</td>
</tr>
</tbody>
</table>
Simulation study: selecting the motif width

Table: Mean error in the width selected by the various model selection approaches.

<table>
<thead>
<tr>
<th></th>
<th>dOOPS</th>
<th>dZOOPS1</th>
<th>dZOOPS2</th>
<th>dTCM1</th>
<th>dTCM2</th>
<th>dTCM3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEME</td>
<td>0.26</td>
<td>0.52</td>
<td>0.57</td>
<td>0.37</td>
<td>0.63</td>
<td>0.10</td>
</tr>
<tr>
<td>Lik</td>
<td>2.55</td>
<td>2.72</td>
<td>2.57</td>
<td>2.74</td>
<td>2.60</td>
<td>2.29</td>
</tr>
<tr>
<td>AIC</td>
<td>0.98</td>
<td>1.33</td>
<td>0.96</td>
<td>1.37</td>
<td>1.04</td>
<td>0.88</td>
</tr>
<tr>
<td>BIC</td>
<td>-0.84</td>
<td>-1.37</td>
<td>-1.23</td>
<td>-1.44</td>
<td>-0.96</td>
<td>-0.25</td>
</tr>
<tr>
<td>likCV</td>
<td>-1.44</td>
<td>-0.71</td>
<td>-0.62</td>
<td>-0.58</td>
<td>-0.37</td>
<td>0.18</td>
</tr>
</tbody>
</table>
Incorporating biological information (prior) into motif finding

- Structurally related families of transcription factors bind to similar sites (Luscombe et al. 2000).
- Eisen (2005) demonstrated that motifs bound by proteins with structurally similar DNA binding domains tend to have similar information content profiles.
- Prior knowledge about the structural class of the mediating transcription factor often translates into constraints on the unknown position weight matrix.
- Sensitivity of motif finding might be enhanced with the use of such constraints.
GAL4 binding

From http://www.cryst.bbk.ac.uk/PPS2/.
cosmo

http://cosmoweb.berkeley.edu/

- cosmo allows constrained detection of motifs with a wide variety of constraints (Keleș et al. (2003), Bembom et al. (2007)).
- Both web server and R implementation.
Types of constraints allowed by cosmo

- Bound constraints on the information content across an intervals. (Mirny and Gelfand (2002) showed that the information content at a given position of a motif is proportional to the number of contacts between the protein and the base pair at that position).

\[ IC_{\text{low}} \leq IC(w) \leq IC_{\text{up}}, \quad w \in l_k. \]

- Shape constraints on the information content profile across an interval. Linear or monotone information content profiles.

- Lower bounds on nucleotide frequencies across an interval.

- Palindromic motifs/intervals. E.g. ACAGCTGT. (If the DNA-binding domains of the transcription factor are homodimeric, the corresponding binding sites will be palindromes of each other.) Can specify two intervals \( k_1 \) and \( k_2 \) that are palindromic w.r.t. each other.

\[ p_{wj} = p(W - w + 1)(4 - j + 1), \quad j = 1, \ldots, \quad w = 1, \ldots, W. \]
Types of constraints allowed by cosmo

- **Submotifs.** (Families of transcription factors are often characterized by occurrence of a certain submotif within the binding site.) E.g., DNA sequences bound by transcription factors with an ETS domain all contain the stretch of GGAA somewhere within the binding site.

Constraints are allowed to be applicable only within certain intervals. cosmo uses model selection techniques to select among a given constraint set.

Under such constraints, M-step of the EM-algorithm is no longer closed form. cosmo employs a Sequential Quadratic Programming (SQP) algorithm to maximize the observed data likelihood.
Dirichlet distribution

- Multivariate generalization of the Beta distribution.
- Conjugate prior of the multinomial distribution.
- A Dirichlet density $\rho$ is a probability density over the set of all probability vectors $p_w$ (i.e., $p_{wj} \geq 0$ and $\sum_{j=1}^{4} p_{wj} = 1$).
- A Dirichlet density has parameters $\alpha = (\alpha_1, \cdots, \alpha_4)$, with $\alpha_i > 0$ and

$$
\rho(p_w) = \frac{\prod_{j=1}^{4} p_{wj}^{\alpha_j-1}}{Z(\alpha)},
$$

where $Z(\alpha)$ is a normalizing constant. The mean value of $p_{wj}$ given a Dirichlet density with parameters $\alpha$ is

$$
E p_{wj} = \frac{\alpha_j}{\sum_{j'=1}^{4} \alpha_{j'}}.
$$
Revisit OOPS model

- Assume a an independent Dirichlet density prior for each position in the position weight matrix. Posterior probability of the $P$:

$$Pr(P | X, \alpha) \propto Pr(X | P, \alpha)Pr(P | \alpha)$$

- Maximum aposteriori estimates of the parameters:

$$P_{\text{MAP}} = \arg\max_P Pr(P | X, \alpha) = \arg\max_P Pr(X | P)Pr(P | \alpha).$$

- Updating formulas in the M-step change to

$$p_{wj}^{\text{new}} = \frac{N_{wj} + (\alpha_j - 1)}{\sum_{j'=1}^{4} (N_{wj'} + (\alpha_{j'} - 1))}.$$ 

- In several papers, MAP estimates are reported as

$$p_{wj}^{\text{new}} = \frac{N_{wj} + \alpha_j}{\sum_{j'=1}^{4} (N_{wj'} + \alpha_{j'})},$$

and $\alpha_j$s are interpreted as pseudo counts. This formula corresponds to setting $p_{wj}$s to their posterior means.
Might have a different Dirichlet priors (or even mixtures of Dirichlet priors) for each position of the PWM and the background distribution. In practice, use a small number for $\alpha_j$.

Practical examples show that these priors are most useful for protein binding sites (over the 20 letter DNA alphabet).
References


References

Chromatin packaging of DNA

DNA is packaged in the form of one or more large macro molecules called chromosomes. In the chromosomes of eukaryotes, the uncondensed DNA exists in a quasi-ordered structure inside the nucleus and wraps around histone proteins. This composite material is called chromatin and is composed of nucleosomes.
Part-II

High-throughput experimental technologies for identifying TF binding sites

- Biotechnologies and high-throughput experiments for studying transcriptional regulation.
- Tiling arrays.
- Modeling issues (HMMs).
High throughput ChIP assay (ChIP-chip): Chromatin immunoprecipitation on DNA chip

- ChIP: Chromatin immunoprecipitation (*in vivo*).
- chip: DNA microarray.
- ChIP-chip: ChIP followed by DNA chip utilization.
ChIP assay

Target protein =
ChIP assay

Target protein =

1. Crosslink DNA and protein *in vivo* by exposing cells to formaldehyde.
1. Crosslink DNA and protein in vivo by exposing cells to formaldehyde.

2. Extract the chromatin from cells and fragment by sonication (~ 500-1000 bps).
ChIP assay

Target protein =

1. Crosslink DNA and protein in vivo by exposing cells to formaldehyde.

2. Extract the chromatin from cells and fragment by sonication (~500-1000 bps).

3. Immunoprecipitate using a target protein-specific antibody.

Selectively binds to the target protein.
ChIP assay

1. Crosslink DNA and protein in vivo by exposing cells to formaldehyde.

2. Extract the chromatin from cells and fragment by sonication (~500-1000 bps).

3. Immunoprecipitate using a target protein-specific antibody.

4. Reverse the cross-links and purify DNA.

Target protein = ![Diagram of target protein binding to DNA](image)

Selectively binds to the target protein.
1. Crosslink DNA and protein in vivo by exposing cells to formaldehyde.

2. Extract the chromatin from cells and fragment by sonication (~ 500-1000 bps).

3. Immunoprecipitate using a target protein-specific antibody.

4. Reverse the cross-links and purify DNA.

5. Find the identity of the isolated DNA fragments using DNA microarrays.
In ChIP-chip experiments measuring nucleosome occupancy, an enzyme called "Micrococcal nuclease" is used to digest nucleosome free regions instead of sonication + immunoprecipitation.
Traditional ChIP assay

- The identity of the DNA fragments isolated in complex with the protein of interest can then be determined by polymerase chain reaction (PCR) using primers specific for the DNA regions that the protein in question is hypothesized to bind.
- One experiment per hypothesized region.
- Can we identify the identity of all the immunoprecipitated regions? Tiling arrays (ChIP-chip), Sequencing (ChIP-Seq).
Tiling arrays

Genome: ACGTAATTTGGTTTTTGTGCCCAAAATAT...
sequence

Probes: ACGTAATTTGGTTTTTGTGCCCAAAATAT...

Tiling Microarray:

Affymetrix tiling arrays:
25bps
~10bps

NimbleGen tiling arrays (18-60 bps with any resolution):
50bps
~50bps

Figure: Can fit entire Drosophila, Yeast, E.coli (and more) genome on a single chip.
NimbleGen Systems

- This is a Madison-based biotech company.
- They specialize in building custom-made, isothermal, high density long oligonucleotide arrays.
- The key is their Maskless Array Synthesizer (MAS) technology.
- MAS system: high density DNA array fabrication instrument comprised of (1) A maskless light projector; (2) A reaction chamber; (3) A personal computer; (4) a DNA synthesizer.
- Arrays are built using photo-mediated synthesis chemistry with the MAS system.
From Niblegen’s website.
From Niblegen’s website.
DNA microarrays rely on the hybridization properties of nucleic acids to monitor DNA or RNA abundance on a genomic scale in different types of cells.

Hybridization refers to the annealing of two nucleic acid strands following the base-pairing rules. E.g. A-T, C-G.

Nucleic acid strands in a duplex can be separated, or denatured, by heating to destroy the hydrogen bonds.

**Figure:** Process of creating a hybrid strand of DNA/RNA.
Figure: Process of creating a hybrid strand of DNA/RNA.

(a) DNA is denatured by heating.
(b) Renaturation on cooling.
(c) Hybridization.
(d) DNA/RNA hybrid.

RNA strand DNA strand

Nucleic Acid Hybridization
Animation: DNA microarrays

http://www.bio.davidson.edu/Courses/genomics/chip/chip.html
ChIP-chip data from tiling arrays

Characteristics:
- Very small sample sizes, typically $n = 2$.
- Spatial signal, bound probes occur in groups (peak structure).
- Spatial correlation due to overlapping probes, highly heterogeneous structure.

Figure: Example ChIP-chip data over 8000 base pairs on the rat genome.
Statistical methods for analyzing ChIP-chip data

Methods need to be typically adapted based on the nature of the ChIP-chip experiments (ChIP-chip for TF binding, Nucleosomes occupancy, Histone modification).

1. Simple one sample or two sample t-test like statistics at the probe level followed by multiplicity adjustment.

2. Empirical Bayes models (modeling probe level data; incorporating peak like structure).

3. Hidden Markov models.
A simplified model for ChIP-chip data

- **Observed data:** Observed (possibly transformed) intensities: $X_1, \ldots, X_N$ over $N$ probes.

- **Unobserved data:** Whether or not a given probe is bound: $Y_1, \ldots, Y_N$. A total of 2 states.
  
  $Y_i = \begin{cases} 1 & \text{if i-th probe is bound}, \\ 0 & \text{o.w.} \end{cases}$

- Assume a 1st order (position) homogeneous Markov chain model for the distribution of $Y_i$s.

  $Pr(Y_i = y_i, \mid Y_1 = y_1, \ldots, Y_{i-1} = y_{i-1}) = Pr(Y_i = y_i, \mid Y_{i-1} = y_{i-1})$.

- Define the transition matrix $A$

  $$A = \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix}, \quad p_{ab} \geq 0, \quad \sum_{b=0}^{1} p_{ab} = 1, a \in \{0, 1\}.$$  

  $p_{00} \equiv$ transition probability from an unbound probe to an unbound probe and so on.

- Assume that $X_i \mid Y_i = 0 \sim \mathcal{N}(\mu_0, \sigma_0^2)$ and $X_i \mid Y_i = 1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$. 
Markov model

$$\begin{align*}
\text{Unbound} & \quad \text{Bound} \\
p_{00} & \quad p_{01} \\
p_{10} & \quad p_{11}
\end{align*}$$
Hidden Markov models (HMMs)

Putting together the observed data $\mathbf{X}$ and the unobserved data $\mathbf{Y}$ described by the Markov chain, we obtain a HMM architecture.

- A hidden Markov model (HMM) is similar to a Markov chain, but is more general and flexible allowing modeling of phenomena that cannot be captured with a regular Markov chain model.
- Main addition: when a state is visited by the Markov chain, the state *emits* an observation from a location (time) independent distribution indexed by the state.
- First order HMM.
Hidden Markov model
Hidden Markov model

$X_1 \rightarrow X_i \rightarrow X_{i+1} \rightarrow X_{i+2} \rightarrow X_N$

$Y_1 \rightarrow Y_i \rightarrow Y_{i+1} \rightarrow Y_{i+2} \rightarrow Y_N$

$Y_1$: Initial state.
Hidden Markov model

\[ Y_i \text{ to } Y_{i+1}: \text{Transition.} \]
Generate $X_i$ based on $Y_i$: Emission.
Hidden Markov model

\[
Pr(X_i \mid Y, X_1, \ldots, X_{i-1}) = Pr(X_i \mid Y_i)
\]

\[
Pr(X \mid Y) = \prod_{i=1}^{N} Pr(X_i \mid Y_i).
\]
Components of an HMM

- A set of $K$ states (for ChIP-chip model: $K = 2$; bound vs. unbound).
- A transition probability matrix.
- An initial state probability vector $\pi$ (essentially probability distribution for $Y_1$). $\pi_a = Pr(Y_1 = a)$, $a \in \{0, 1\}$.
- Emission distributions: $f_a(.)$ : emission distribution in state $a$, $a \in \{0, 1\}$.

Parameters of an HMM: $\pi$, $A$, parameters of $f_0$ and $f_1$. Denote this collection by $\Psi$. 
Calculations in the HMM framework

- Given the parameters $\Psi$, calculate
  
  $$Pr(X_1, \cdots, X_N \mid \Psi).$$

- Given the observed data $X_1, \cdots, X_N$, find the most likely hidden sequence of states
  
  $$\arg\max_{y_1, \cdots, y_N} Pr(Y_1 = y_1, \cdots, Y_N = y_n \mid X_1, \cdots, X_N).$$

- Given the observed data $X_1, \cdots, X_N$, estimate the parameters, $\Psi$. 
What is special about these calculations?

They are (seem) pretty difficult, i.e., computationally heavy? \textit{Order}: 
\[ 2K \times K^N \text{ (e.g. } K = 2). \]
Assume \( N = 10 \). Need to sum over all possible paths to evaluate the likelihood. There are \( 2^{10} \) of these. In general, \( K^N \).

\[
Pr(X_1, \cdots, X_{10} \mid \psi) = \sum_{y_1=0}^{1} \cdots \sum_{y_{10}=0}^{1} Pr(X_1, \cdots, X_{10} \mid Y_1 = y_1, \cdots, Y_{10} = y_{10}) \times Pr(Y_1 = y_1, \cdots, Y_{10} = y_{10})
\]

Note: Under the 1st order HMM (suppressing \( \psi \)):

\[
Pr(X_1, \cdots, X_{10} \mid Y_1 = y_1, \cdots, Y_{10} = y_{10}) = \prod_{i=1}^{10} Pr(X_i \mid Y_i = y_i) = \prod_{i=1}^{10} f_{y_i}(X_i)
\]

\[
Pr(Y_1 = y_1, \cdots, Y_{10} = y_{10}) = Pr(Y_1 = y_1) \prod_{i=1}^{10} Pr(Y_i = y_i \mid Y_{i-1} = y_{i-1}) = \pi_{y_1} \prod_{i=2}^{10} p_{y_{i-1}y_i}.
\]
Computational tricks for HMM computations

- Forward algorithm.
- Backward algorithm.
- Viterbi algorithm.
Forward algorithm: efficient computation for $Pr(\mathbf{X} \mid \Psi)$

- Define $\alpha(i, a) \equiv Pr(X_1 = x_1, \cdots, X_i = x_i, Y_i = a)$: joint probability of the sequence of observations up to and including $i$th position and the state of the $i$th position.

- Once we know $\alpha(N, a)$ for all $a$, then

$$Pr(\mathbf{X} \mid \Psi) = \sum_{a=0}^{1} \alpha(N, a).$$

- Main idea: Compute $\alpha(N, a)$ inductively.
Forward algorithm

- **Initialization:** \( \alpha(1, a) = \pi_a f_a(X_1) \).

- **\( \alpha(i + 1, a) \):**

  \[
  \alpha(i + 1, a) = \Pr(X_1 = x_1, \cdots, X_i = x_i, X_{i+1} = x_{i+1}, Y_{i+1} = a)
  \]

  \[
  = \sum_{b=0}^{1} Pr(X_1 = x_1, \cdots, X_i = x_i, X_{i+1} = x_{i+1}, Y_i = b, Y_{i+1} = b)
  \]

  \[
  = \sum_{b=0}^{1} Pr(X_1 = x_1, \cdots, X_i = x_i \mid X_{i+1} = x_{i+1}, Y_i = b, Y_{i+1} = a) 
  \times Pr(X_{i+1} = x_{i+1} \mid Y_i = b, Y_{i+1} = a) Pr(Y_i = b, Y_{i+1} = a)
  \]

  \[
  = \sum_{b=0}^{1} \frac{Pr(X_1 = x_1, \cdots, X_i = x_i, Y_i = b)}{Pr(Y_i = b)} f_a(X_{i+1}) Pr(Y_i = b, Y_{i+1} = a)
  \]

  \[
  = \sum_{b=0}^{1} Pr(X_1 = x_1, \cdots, X_i = x_i, Y_i = b)f_a(X_{i+1}) Pr(Y_{i+1} = a \mid Y_i = b)
  \]

  \[
  = \sum_{b=0}^{1} \alpha(i, b) f_a(X_{i+1}) p_{ba}
  \]
Forward algorithm

Compute $\alpha(1, .), \alpha(2, .), \cdots, \alpha(N, .)$ in a forward manner. \textit{Order: } $N \times K^2. \ (K = 2)$
Backward algorithm

- Define $\beta(i, a) = Pr(X_{i+1} = x_{i+1}, \ldots, X_N = x_N \mid Y_i = a)$, $1 \leq i \leq N - 1$.
- Define $\beta(N, a) = 1 \forall a$.
- Note that

$$Pr(X_1, \ldots, X_N) = \sum_{a=0}^{1} Pr(X_1, \ldots, X_N \mid Y_1 = a) Pr(Y_1 = a)$$

$$= \sum_{a=0}^{1} Pr(X_2, \ldots, X_N \mid Y_1 = a, X_1) Pr(X_1 \mid Y_1 = a) \pi_a$$

$$= \sum_{a=0}^{1} \left( Pr(X_2, \ldots, X_N \mid Y_1 = a, X_1) \frac{Pr(X_1 \mid Y_1 = a) \pi_a}{Pr(X_2, \ldots, X_N \mid Y_1 = a)} \right)$$

$$= \sum_{a=0}^{1} \beta(1, a) f_a(X_1) \pi_a.$$

- $\beta(i - 1, a) = \sum_{b=0}^{1} p_{ab} f_b(X_i) \beta(i, b)$. [Ex. Show!]
Viterbi algorithm

Find the most likely hidden state sequence \( y_1, \cdots, y_N \) such that

\[
\arg\max_{y_1, \cdots, y_N} Pr(Y_1 = y_1, \cdots, Y_N = y_N \mid X_1, \cdots, X_N).
\]

Viterbi algorithm is a dynamic programming algorithm. It has two steps:

- Find \( \max_{y_1, \cdots, y_N} Pr(Y_1 = y_1, \cdots, Y_N = y_N \mid X_1, \cdots, X_N) \).
- Backtrack \( y_1, \cdots, y_N \) that realizes this maximum.
Viterbi algorithm

- Note that

$$\arg\max_y Pr(Y = y \mid X) = \arg\max_y \frac{Pr(Y = y, X)}{Pr(X)} = \arg\max_y Pr(Y = y, X)$$

- Define, for arbitrary $i$ and $a$,

$$\delta(i, a) = \max_{Y_1, \ldots, Y_{i-1}} Pr(Y_1 = y_1, \ldots, Y_{i-1} = y_{i-1}, Y_i = a, X_1 = x_1, \ldots, X_i = x_i)$$

This is the maximum probability of always ending in state $a$ at position $i$ and having the observed sequence $x_1, \ldots, x_i$.

- $\delta(1, a) = Pr(Y_1 = a, X_1 = x_1)$. 

Stat 992 (877) (Spring 08)
Viterbi algorithm

First compute, $\delta(i, a)$ inductively for all $i$ and then recover the sequence that gives the largest $\delta(N, a)$.

- **Initialize:** $\delta(1, a) = \pi_a f_a(X_1)\ , \ a \in \{0, 1\}$.
- **Induction:**
  $$\delta(i, b) = \max_{a \in \{0, 1\}} \delta(i - 1, a) \pi_{ab} f_b(X_i), \quad 2 \leq i \leq N, \quad b \in \{0, 1\}.$$ 
- **Recover $y_i$s:**
  - Define
    $$y_N = \arg\max_{a \in \{0, 1\}} \delta(N, a).$$
  Then $y_N$ is the final state in the state sequence required,
  - The remaining $y_i$ for $i \leq N - 1$ are found recursively by defining
    $$y_i = \arg\max_{a \in \{0, 1\}} \delta(i, a) \pi_{ay_{i+1}}.$$ 
  - If $\arg\max$ is not unique, we arbitrarily take one values of $a$ giving the maximum.
Fitting an HMM with the EM algorithm

- The EM algorithm for HMMs is also known as the Baum-Welch algorithm (published earlier than the EM algorithm of Dempster, Laird and Rubin (1977)).

**Complete data log-likelihood:**

\[
\log \Pr(X, Y \mid \Psi) = \log \left[ \prod_{i=1}^{N} \prod_{a=0}^{1} f_a(X_i)^{I(Y_i=a)} \right] \times \left[ \prod_{i=2}^{N} \prod_{a=0}^{1} \prod_{b=0}^{1} p_{ab}^{I(Y_i=a, Y_{i+1}=b)} \right] \times \left[ \prod_{a=0}^{1} \pi^{I(Y_1=a)} \right].
\]
Fitting an HMM with the EM algorithm

Expected complete data log-likelihood (Q-function).

\[
E(\log Pr(X, Y \mid \psi) \mid X, \psi^{\text{old}}) =
\sum_{i=1}^{N} \sum_{a=0}^{1} Pr(Y_i = a \mid X, \psi^{\text{old}}) \log \left[ \frac{1}{\sqrt{2\pi\sigma_a^2}} \exp \left( -\frac{(X_i - \mu_a)^2}{2\sigma_a^2} \right) \right] +
\sum_{i=2}^{N} \sum_{a=0}^{1} \sum_{b=0}^{1} Pr(Y_i = a, Y_{i+1} = b \mid X, \psi^{\text{old}}) \log p_{ab} +
\sum_{a=0}^{1} Pr(Y_1 = a \mid X, \psi^{\text{old}}) \log \pi_a
\]
Fitting an HMM with the EM algorithm: E-step

\[ Pr(Y_i = a \mid \mathbf{X}, \psi^{\text{old}}) = \frac{Pr(X_1, \ldots, X_i, Y_i = a) Pr(X_{i+1}, \ldots, X_N \mid Y_i = a)}{Pr(X_1, \ldots, X_N)}. \]

\[ \gamma_i(a) = \frac{\alpha(i, a) \beta(i, a)}{\sum_{a' = 0}^1 \alpha(i, a') \beta(i, a')} \]

\[ \gamma_1(a) = \frac{\alpha(i, a) \beta(i, a)}{\sum_{a' = 0}^1 \alpha(N, a')} \]

\[ Pr(Y_1 = a \mid \mathbf{X}, \psi^{\text{old}}) = \gamma_1(a). \]

Define \( \eta_{i+1}(a, b) = Pr(Y_i = a, Y_{i+1} = b \mid \mathbf{X}, \psi^{\text{old}}). \)

\[ \eta_{i+1}(a, b) = \frac{\alpha(i, a)p_{ab}f_b(X_{i+1})\beta(i + 1, b)}{\sum_{a' = 0}^1 \sum_{b' = 0}^1 \alpha(i, a')p_{a'b'}f_{b'}(X_{i+1})\beta(i + 1, b')} \]
Fitting an HMM with the EM algorithm: M-step

\[ \max_{\psi} E[\log(Pr(X, Y | \psi)) | X, \psi^{old}] \]

s.t. \[ \sum_{a=0}^{1} \pi_a = 1 \]

\[ \sum_{b=0}^{1} p_{ab} = 1, \quad \forall a \in \{0, 1\}. \]
Fitting an HMM with the EM algorithm: M-step

- Initial probability estimates:
  \[ \hat{\pi}_a = \gamma_1(a) \]

- Transition matrix estimates:
  \[ \hat{p}_{ab} = \frac{\sum_{i=1}^{N-1} \eta_{i+1}(a, b)}{\sum_{i=1}^{N-1} \gamma_i(a)} \]

- Estimates of the parameters of the emission distributions:
  \[ \hat{\mu}_a = \frac{\sum_{i=1}^{N} \gamma_i(a) X_i}{\sum_{i=1}^{N} \sum_{a'=0}^{1} \gamma_i(a')}, \quad a \in \{0, 1\} \]
  \[ \hat{\sigma}^2_a = \frac{\sum_{i=1}^{N} \gamma_i(a)(X_i - \hat{\mu}_a)^2}{\sum_{i=1}^{N} \sum_{a'=0}^{1} \gamma_i(a')}, \quad a \in \{0, 1\} \]
Some extensions on the basic HMM architecture

- Number of states are problem dependent: $K$.
- Emission distributions (discrete, non-normal).
- Nonhomogeneous HMMs. Transition probabilities depend on the actual location, perhaps, through a covariate.
- Extensions on the duration probabilities. In the current setting:

$$Pr(d \text{ consecutive observations in state } a) = p_{aa}^{d-1}(1 - p_{aa}),$$

- Duration in a state is geometrically distribution.
- Can choose a duration distribution to reflect the underlying process (hidden semi-Markov models).
- Emissions can depend on current state and (k-1) previous states or even observations of the some of the previous states.
Final remarks

- HMMs have a wide variety of applications, from gene finding to gene expression modeling, in computational biology.
- As model gets more complex, run time and memory requirements increase.
- The implementation includes multiplying very small numbers together. Special care is often required to avoid numerical problems (e.g., scaling of $\alpha$ and $\beta$s properly).
Autocorrelation plots from ChIP-chip data

(a) Length = 50 bps  
Resolution = 38 bps  
Overlap = 12 bps

(b) Length = 45 bps  
Resolution = 22 bps  
Overlap = 23 bps

(c) Length = 50 bps  
Resolution = 100 bps  
Spacing = 50 bps
Moving average (sliding window) statistic

- Can be defined over a fixed number of probes or fixed genomic distance.
- Let $Y_1, Y_2, ..., Y_N$ denote measurements on the $N$ probes.
- Let $w_i$ define a window size of $2w_i + 1$ ($w_i$ probes to the left and right of $i$-th probe).
- Moving average statistics for probe $i$, $(T_i)$:

$$T_i = \frac{1}{2w_i + 1} \sum_{j=i-w_i}^{i+w_i} Y_i$$
Moving average (sliding window) statistic

- **Standardized moving average statistic**

  \[
  S_i = \frac{T_i}{\sqrt{\text{var}(T_i)}}
  \]

  \[
  \text{var}(T_i) = \frac{1}{(2w_i + 1)^2} \left( (2w_i + 1)\sigma^2 + \sum_{j=i-w_i}^{i+w_i} \sum_{k \neq j} \text{cov}(Y_j, Y_k) \right)
  \]

- **Standard practice of using moving average statistics relies on:**
  - estimating \( \sigma^2 \) based on the observations that represent lower half of the unbound distribution.
  - ignoring the covariance term in \( \text{var}(T_i) \).
  - obtaining a null distribution under the hypothesis of no binding at probe \( i \) by permutation or Gaussian approximation (assuming exchangeability of \( Y_i \)'s).
Problems with ignoring the correlation structure

- Increase in the number of false positives.

(d) log₂ ratios

(e) negative log p-values
Problems with ignoring the correlation structure

- Misspecification of distribution for $S_i$.
  
  Histogram of p-values

(f) As expected.

(g) Ignoring the correlation structure.
Recall: probability integral transformation

- $X$ is continuous with cdf $F_X(.)$.
- Define random variable $Y = F_X(X)$.
- $Y$ is uniformly distributed on $[0, 1]$.
- Why:

\[
Pr(Y \leq y) = Pr(F_X(X) \leq y) = Pr(F_X^{-1}F_X(x) \leq F_X^{-1}(y)) = Pr(X \leq F_X(y)) = F_X(F_X^{-1}(y)) = y.
\]

- If $F_X$ is not strictly increasing, i.e., constant in some intervals, then define $F_X^{-1}(y) = \inf\{x : F_X(x) \geq y\}$.
- Apply with p-values: $H_0 : \theta = \theta_0$ vs. $H_1 : \theta \neq \theta_0$.
  
  p-value = $p_\theta(x) = Pr_{\theta_0}(T(X) \geq T(x))$. 

Problems with ignoring the correlation structure

- Misspecification of distribution for $S_i$.
  QQ plots of the test statistics.

(h) Normality assumption is valid. (i) Ignoring correlation.
Estimating the correlation structure

- Correlation, Moving Average, Robust and Rapid method on Tiling array (CMARRT): A fast empirical method using $\hat{\rho}(k)$.

$$\hat{\rho}(k) = \frac{\sum_{t=1}^{T-k} (Y_t - \bar{Y})(Y_{t+k} - \bar{Y})}{\sum_{t=1}^{T} (Y_t - \bar{Y})^2}, \quad \hat{\text{cov}}(Y_j, Y_{j+k}) = \hat{\rho}(k)\hat{\sigma}^2$$

Algorithm

- Remove top $M\%$ of outlying probes.
- Identify segments of at least N consecutive probes. Regions flanking large gaps or repeat masked regions are treated as two separate segments.
- For each segment $j$, compute $\hat{\rho}_j(k)$.
- For any lag $k$, let $\hat{\rho}(k)$ be the average $\hat{\rho}_j(k)$ over $j$. 
Simulation studies

A Hidden Markov model simulation

- Data is simulated from an HMM with explicit duration distribution to introduce direct dependencies at probe level observations.
- Unbound regions: $f_{N_i}(Y_1, Y_2, ... Y_{d_1}) \sim MVN(0, \Sigma_N)$.
- Bound regions: $f_{B_i}(Y_1, Y_2, ... Y_{d_1}) \sim MVN(\mu, \Sigma_B), \mu > 0$

Compare the performances of CMARRT, Indep and TileMap (Ji et al. (2005)). A probe is declared to be bound using false discovery rate (FDR) control.
Results of the HMM simulation

- **Sensitivity (peaksize=10)**
  - FDR Rate
    - 0.4
    - 0.6
    - 0.8
    - 1.0
- **Specificity (peaksize=10)**
  - Duration HMM
  - CMARRT
  - Indep
  - TileMap

- **Sensitivity (peaksize=20)**
  - FDR Rate
    - 0.4
    - 0.6
    - 0.8
    - 1.0
- **Specificity (peaksize=20)**
  - Duration HMM
  - CMARRT
  - Indep
  - TileMap

Stat 992 (877) (Spring 08)
Case study: ZNF217 ChIP-chip data

Under correlation structure

Under independence

Under correlation structure

Under independence
Other common applications of tiling arrays

- Expression studies.
- Copy number change studies.
References


The autocorrelation function (ACF) of a stochastic process describes the correlation between the process at different points in time. An estimate of $k$-th order autocorrelation is

$$\frac{\sum_{i=1}^{N-k} (Y_i - \bar{Y})(Y_{i+k} - \bar{Y})}{\sum_{i=1}^{N} (Y_i - \bar{Y})^2}.$$