Expectation Maximization and Gibbs Sampling

Lecture 1 - Introduction

Lecture 2 - Hashing and BLAST

Lecture 3 - Combinatorial Motif Finding

Lecture 4 - Statistical Motif Finding

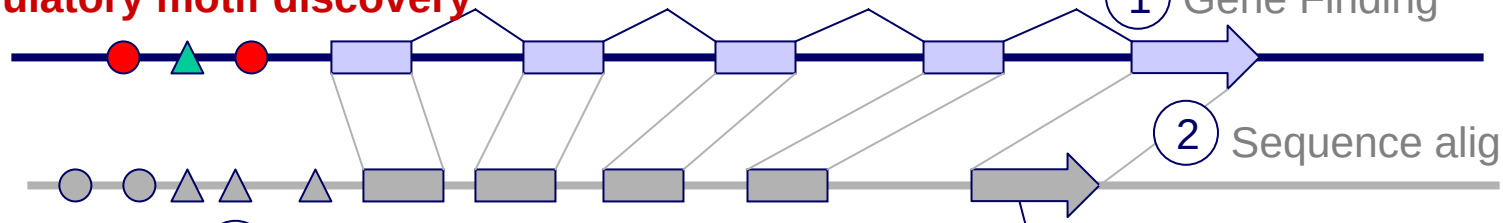
Challenges in Computational Biology

4 Genome Assembly

5 **Regulatory motif discovery**

1 Gene Finding

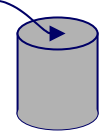
DNA



2 Sequence alignment

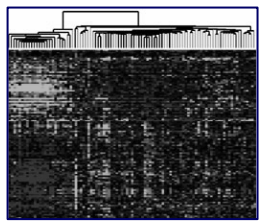
6 Comparative Genomics

7 Evolutionary Theory

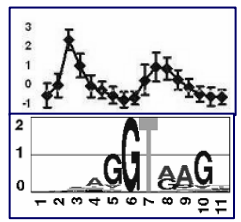


3 Database lookup

8 Gene expression analysis

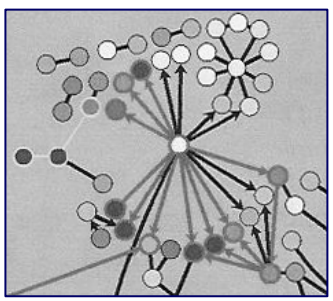


9 Cluster discovery



10 Gibbs sampling

11 Protein network analysis

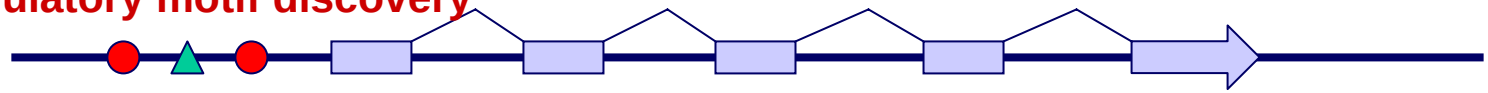


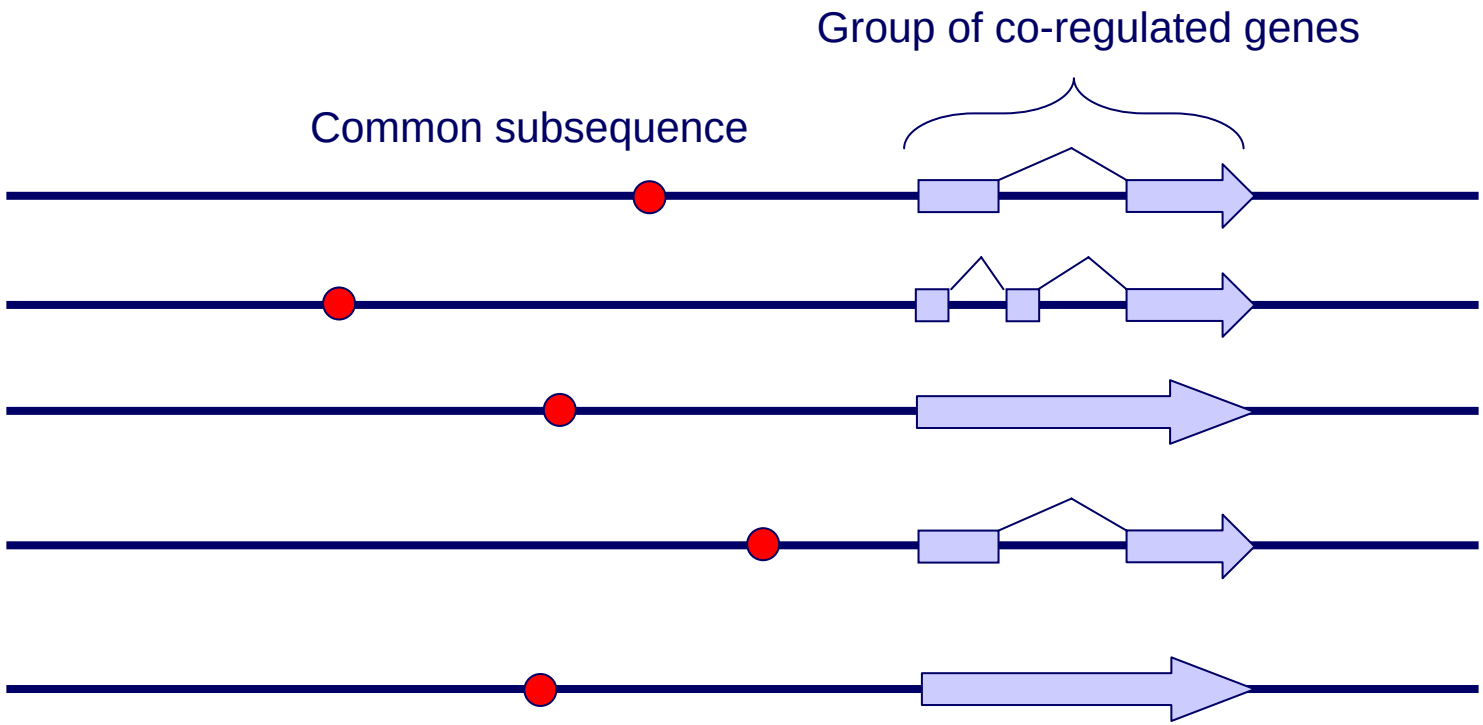
12 Regulatory network inference

13 Emerging network properties

Challenges in Computational Biology

5 Regulatory motif discovery





Overview

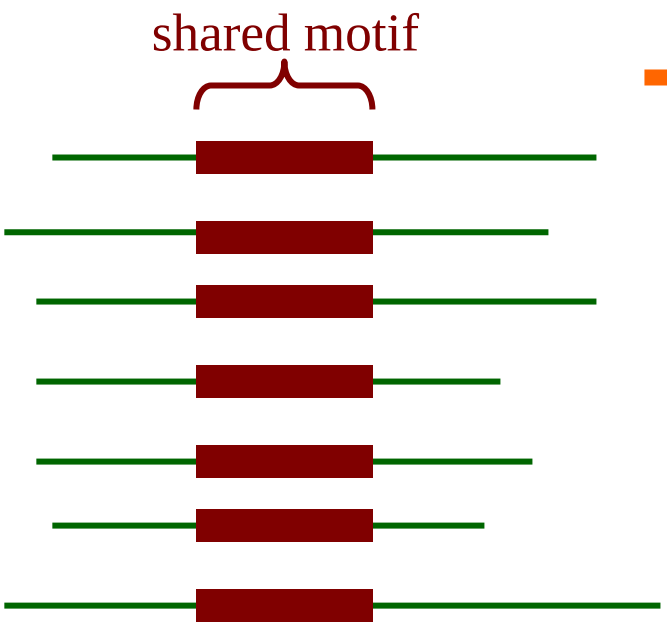
- Introduction
  - Bio review: Where do ambiguities come from?
  - Computational formulation of the problem
- Combinatorial solutions
  - Exhaustive search
  - Greedy motif clustering
  - Wordlets and motif refinement
- Probabilistic solutions
  - **Expectation maximization**
  - Gibbs sampling

Sequence Motifs

- what is a sequence *motif* ?
  - a sequence pattern of biological significance
- examples
  - protein binding sites in DNA
  - protein sequences corresponding to common functions or conserved pieces of structure

Motifs and Profile Matrices

- given a set of aligned sequences, it is straightforward to construct a profile matrix characterizing a motif of interest

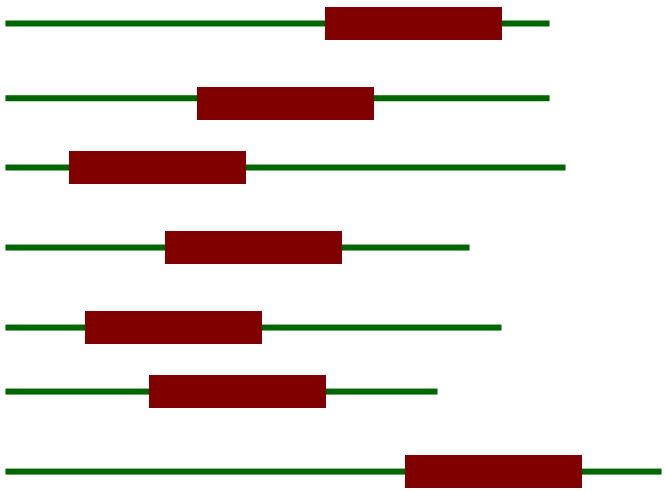


sequence positions

	1	2	3	4	5	6	7	8
A	0.1	0.3	0.1	0.2	0.2	0.4	0.3	0.1
C	0.5	0.2	0.1	0.1	0.6	0.1	0.2	0.7
G	0.2	0.2	0.6	0.5	0.1	0.2	0.2	0.1
T	0.2	0.3	0.2	0.2	0.1	0.3	0.3	0.1

Motifs and Profile Matrices

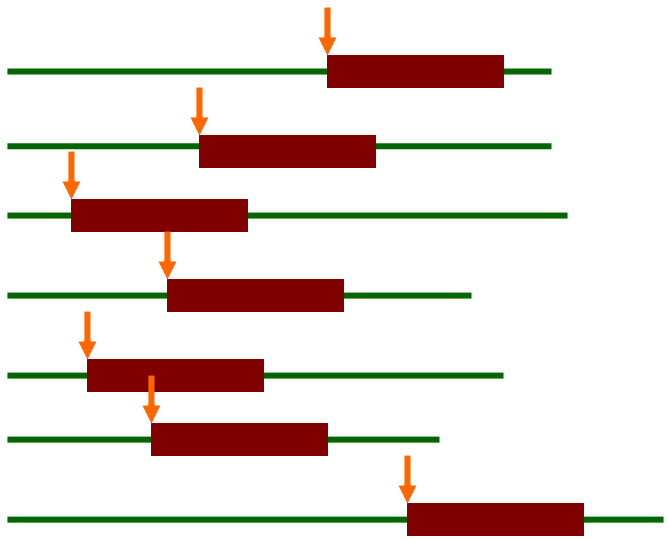
- how can we construct the profile if the sequences aren't aligned?
  - in the typical case we don't know what the motif looks like



- use an Expectation Maximization (EM) algorithm

The EM Approach

- EM is a family of algorithms for learning probabilistic models in problems that involve *hidden state*
- in our problem, the hidden state is where the motif starts in each training sequence



The MEME Algorithm

- Bailey & Elkan, 1993
- uses EM algorithm to find multiple motifs in a set of sequences
- first EM approach to motif discovery: Lawrence & Reilly 1990

Representing Motifs

- a motif is assumed to have a fixed width, W
- a motif is represented by a matrix of probabilities: p_{ck} represents the probability of character c in column k
- example: DNA motif with W=3

$$p = \begin{array}{c|ccc} & 1 & 2 & 3 \\ A & 0.1 & 0.5 & 0.2 \\ C & 0.4 & 0.2 & 0.1 \\ G & 0.3 & 0.1 & 0.6 \\ T & 0.2 & 0.2 & 0.1 \end{array}$$

Representing Motifs

- we will also represent the “background” (i.e. outside the motif) probability of each character
- p_{c0} represents the probability of character *c* in the background
- example:

$$p_0 = \begin{matrix} \texttt{A} & \texttt{0.26} \\ \texttt{C} & \texttt{0.24} \\ \texttt{G} & \texttt{0.23} \\ \texttt{T} & \texttt{0.27} \end{matrix}$$

Basic EM Approach

- the element Z_{ij} of the matrix Z represents the probability that the motif starts in position *j* in sequence *I*
- example: given 4 DNA sequences of length 6, where *W*=3

$$Z = \begin{array}{c} \\ \texttt{seq1} \\ \texttt{seq2} \\ \texttt{seq3} \\ \texttt{seq4} \end{array} \begin{array}{cccc} \texttt{1} & \texttt{2} & \texttt{3} & \texttt{4} \\ \texttt{0.1} & \texttt{0.1} & \texttt{0.2} & \texttt{0.6} \\ \texttt{0.4} & \texttt{0.2} & \texttt{0.1} & \texttt{0.3} \\ \texttt{0.3} & \texttt{0.1} & \texttt{0.5} & \texttt{0.1} \\ \texttt{0.1} & \texttt{0.5} & \texttt{0.1} & \texttt{0.3} \end{array}$$

Basic EM Approach

given: length parameter W, training set of sequences

set initial values for p

do

re-estimate Z from p (E –step)

re-estimate p from Z (M-step)

until change in $p < \varepsilon$

return: p, Z

Basic EM Approach

- we'll need to calculate the probability of a training sequence given a hypothesized starting position:

$$\Pr(X_i \mid Z_{ij} = 1, p) = \underbrace{\prod_{k=1}^{j-1} p_{c_k,0}}_{\text{before motif}} \underbrace{\prod_{k=j}^{j+W-1} p_{c_k,k-j+1}}_{\text{motif}} \underbrace{\prod_{k=j+W}^{L} p_{c_k,0}}_{\text{after motif}}$$

X_i is the *i*th sequence

Z_{ij} is 1 if motif starts at position *j* in sequence *i*

c_k is the character at position *k* in sequence *i*

Example

$X_i =$ **G C T G T A G**

$p =$

	0	1	2	3
A	0.25	0.1	0.5	0.2
C	0.25	0.4	0.2	0.1
G	0.25	0.3	0.1	0.6
T	0.25	0.2	0.2	0.1

$$\Pr(X_i \mid Z_{i3} = 1, p) =$$

$$p_{\mathrm{G},0} \times p_{\mathrm{C},0} \times p_{\mathrm{T},1} \times p_{\mathrm{G},2} \times p_{\mathrm{T},3} \times p_{\mathrm{A},0} \times p_{\mathrm{G},0} =$$

$$0.25 \times 0.25 \times 0.2 \times 0.1 \times 0.1 \times 0.25 \times 0.25$$

The E-step: Estimating Z

- to estimate the starting positions in Z at step t

$$Z_{ij}^{(t)} = \frac{\Pr(X_i \mid Z_{ij} = 1, p^{(t)}) \Pr(Z_{ij} = 1)}{\sum_{k=1}^{L-W+1} \Pr(X_i \mid Z_{ik} = 1, p^{(t)}) \Pr(Z_{ik} = 1)}$$

- this comes from Bayes' rule applied to

$$\Pr(Z_{ij} = 1 \mid X_i, p^{(t)})$$

The E-step: Estimating Z

- assume that it is equally likely that the motif will start in any position

$$Z_{ij}^{(t)} = \frac{\Pr(X_i \mid Z_{ij} = 1, p^{(t)}) \Pr(Z_{ij} = 1)}{\sum_{k=1}^{L-W+1} \Pr(X_i \mid Z_{ik} = 1, p^{(t)}) \Pr(Z_{ik} = 1)}$$

Example: Estimating Z

$X_i =$ **G C T G T A G**

$p =$

	0	1	2	3
A	0.25	0.1	0.5	0.2
C	0.25	0.4	0.2	0.1
G	0.25	0.3	0.1	0.6
T	0.25	0.2	0.2	0.1

$$Z_{i1} = 0.3 \times 0.2 \times 0.1 \times 0.25 \times 0.25 \times 0.25 \times 0.25$$

$$Z_{i2} = 0.25 \times 0.4 \times 0.2 \times 0.6 \times 0.25 \times 0.25 \times 0.25$$

$$\vdots$$

- then normalize so that $\sum_{j=1}^{L-W+1} Z_{ij} = 1$

The M-step: Estimating *p*

- recall $p_{c,k}$ represents the probability of character *c* in position *k* ; values for position 0 represent the background

$$p_{c,k}^{(t+1)} = \frac{n_{c,k} + d_{c,k}}{\sum_b (n_{b,k} + d_{b,k})}$$

pseudo-counts

$$n_{c,k} = \begin{cases} \sum_i \sum_{\{j | X_{i,j+k-1} = c\}} Z_{ij} & k > 0 \\ n_c - \sum_{j=1}^{W} n_{c,j} & k = 0 \end{cases}$$

total # of c's in data set

Example: Estimating *p*

A C A G C A

$$Z_{1,1} = 0.1, Z_{1,2} = 0.7, Z_{1,3} = 0.1, Z_{1,4} = 0.1$$

A G G C A G

$$Z_{2,1} = 0.4, Z_{2,2} = 0.1, Z_{2,3} = 0.1, Z_{2,4} = 0.4$$

T C A G T C

$$Z_{3,1} = 0.2, Z_{3,2} = 0.6, Z_{3,3} = 0.1, Z_{3,4} = 0.1$$

$$p_{\mathrm{A},1} = \frac{Z_{1,1} + Z_{1,3} + Z_{2,1} + Z_{3,3} + 1}{Z_{1,1} + Z_{1,2} \ldots + Z_{3,3} + Z_{3,4} + 4}$$

The EM Algorithm

- EM converges to a local maximum in the likelihood of the data given the model:

$$\prod_i \Pr(X_i \mid p)$$

- usually converges in a small number of iterations
- sensitive to initial starting point (i.e. values in p)

Overview

- Introduction
  - Bio review: Where do ambiguities come from?
  - Computational formulation of the problem
- Combinatorial solutions
  - Exhaustive search
  - Greedy motif clustering
  - Wordlets and motif refinement
- Probabilistic solutions
  - Expectation maximization
  - **MEME extensions**
  - Gibbs sampling

MEME Enhancements to the Basic EM Approach

- MEME builds on the basic EM approach in the following ways:
  - trying many starting points
  - not assuming that there is exactly one motif occurrence in every sequence
  - allowing multiple motifs to be learned
  - incorporating Dirichlet prior distributions

Starting Points in MEME

- for every distinct subsequence of length *W* in the training set
  - derive an initial *p* matrix from this subsequence
  - run EM for 1 iteration
- choose motif model (i.e. *p* matrix) with highest likelihood
- run EM to convergence

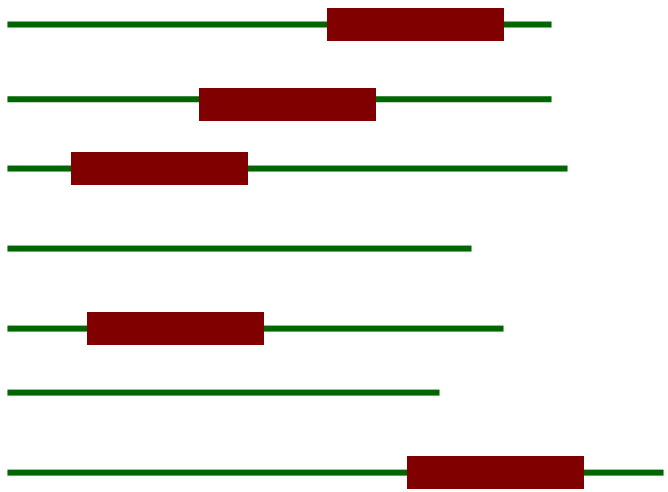
Using Subsequences as Starting Points for EM

- set values corresponding to letters in the subsequence to X
- set other values to $(1-X)/(M-1)$ where M is the length of the alphabet
- example: for the subsequence TAT with X=0.5

$$p = \begin{array}{cccc} & 1 & 2 & 3 \\ A & 0.17 & 0.5 & 0.17 \\ C & 0.17 & 0.17 & 0.17 \\ G & 0.17 & 0.17 & 0.17 \\ T & 0.5 & 0.17 & 0.5 \end{array}$$

The ZOOPS Model

- the approach as we've outlined it, assumes that each sequence has exactly one motif occurrence per sequence; this is the OOPS model
- the ZOOPS model assumes *zero or one* occurrences per sequence



E-step in the ZOOPS Model

- we need to consider another alternative: the *i*th sequence doesn't contain the motif
- we add another parameter (and its relative)

λ

- prior prob that any position in a sequence is the start of a motif

$\gamma = (L - W + 1)\lambda$

- prior prob of a sequence containing a motif

E-step in the ZOOPS Model

$$Z_{ij}^{(t)} = \frac{\Pr(X_i \mid Z_{ij} = 1, p^{(t)})\lambda^{(t)}}{\Pr(X_i \mid Q_i = 0, p^{(t)})(1-\gamma^{(t)}) + \sum_{k=1}^{L-W+1} \Pr(X_i \mid Z_{ik} = 1, p^{(t)})\lambda^{(t)}}$$

- here Q_i is a random variable that takes on 0 to indicate that the sequence doesn't contain a motif occurrence

$$Q_i = \sum_{j=1}^{L-W+1} Z_{i,j}$$

M-step in the ZOOPS Model

- update *p* same as before
- update λ, γ as follows

$$\lambda^{(t+1)} = \frac{\gamma^{(t+1)}}{(L-W+1)} = \frac{1}{n(L-W+1)} \sum_{i=1}^{n} \sum_{j=1}^{m} Z_{i,j}^{(t)}$$

- average of $Z_{i,j}^{(t)}$ across all sequences, positions

The TCM Model

- the TCM (two-component mixture model) assumes *zero or more* motif occurrences per sequence

Likelihood in the TCM Model

- the TCM model treats each length *W* subsequence independently
- to determine the likelihood of such a subsequence:

$$\Pr(X_{ij} \mid Z_{ij} = 1, p) = \prod_{k=j}^{j+W-1} p_{c_k,k-j+1}$$

assuming a motif starts there

$$\Pr(X_{ij} \mid Z_{ij} = 0, p) = \prod_{k=j}^{j+W-1} p_{c_k,0}$$

assuming a motif doesn't start there

E-step in the TCM Model

$$Z_{ij}^{(t)} = \frac{\Pr(X_{i,j} \mid Z_{ij} = 1, p^{(t)})\lambda^{(t)}}{\underbrace{\Pr(X_{i,j} \mid Z_{ij} = 0, p^{(t)})(1-\lambda^{(t)})}_{\text{subsequence isn't a motif}} + \underbrace{\Pr(X_{i,j} \mid Z_{ij} = 1, p^{(t)})\lambda^{(t)}}_{\text{subsequence is a motif}}}$$

- M-step same as before

Finding Multiple Motifs

- basic idea: discount the likelihood that a new motif starts in a given position if this motif would overlap with a previously learned one
- when re-estimating Z_{ij}, multiply by $\Pr(V_{ij} = 1)$

$$V_{ij} = \begin{cases} 1, & \text{no previous motifs in } [X_{i,j},..., X_{i,j+w-1}] \\ 0, & \text{otherwise} \end{cases}$$

- V_{ij} is estimated using Z_{ij} values from previous passes of motif finding

Overview

- Introduction
  - Bio review: Where do ambiguities come from?
  - Computational formulation of the problem
- Combinatorial solutions
  - Exhaustive search
  - Greedy motif clustering
  - Wordlets and motif refinement
- Probabilistic solutions
  - Expectation maximization
  - MEME extensions
  - **Gibbs sampling**

Gibbs Sampling

- a general procedure for sampling from the joint distribution of a set of random variables $\Pr(U_1 \ldots U_n)$ by iteratively sampling from $\Pr(U_j \mid U_1 \ldots U_{j-1}, U_{j+1} \ldots U_n)$ for each j
- application to motif finding: Lawrence et al. 1993
- can view it as a stochastic analog of EM for this task
- less susceptible to local minima than EM

Gibbs Sampling Approach

- in the EM approach we maintained a distribution Z_i over the possible motif starting points for each sequence
- in the Gibbs sampling approach, we'll maintain a specific starting point for each sequence a_i but we'll keep resampling these

Gibbs Sampling Approach

given: length parameter *W*, training set of sequences

choose random positions for *a*

do

pick a sequence X_i

estimate *p* given current motif positions *a* (update step)

(using all sequences but X_i)

sample a new motif position a_i for X_i (sampling step)

until convergence

return: *p*, *a*

Sampling New Motif Positions

- for each possible starting position, $a_i = j$, compute a weight

$$A_j = \frac{\prod_{k=j}^{j+W-1} p_{c_k,k-j+1}}{\prod_{k=j}^{j+W-1} p_{c_k,0}}$$

- randomly select a new starting position a_i according to these weights

Gibbs Sampling (AlignACE)

- Given:
  - $x^1, \ldots, x^N$,
  - motif length K,
  - background B,

$$\sum_{i=1}^{N} \sum_{k=1}^{K} \log \frac{M(k, x^i_{a_i+k})}{B(x^i_{a_i+k})}$$

- Find:
  - Model M
  - Locations $a_1, \ldots, a_N$ in $x^1, \ldots, x^N$

Maximizing log-odds likelihood ratio:

Gibbs Sampling (AlignACE)

- AlignACE: first statistical motif finder
- BioProspector: improved version of AlignACE

Algorithm (sketch):

1. Initialization:
   a. Select random locations in sequences x^1, …, x^N
   b. Compute an initial model M from these locations

2. Sampling Iterations:
   a. Remove one sequence x^i
   b. Recalculate model
   c. Pick a new location of motif in x^i according to probability the location is a motif occurrence

Gibbs Sampling (AlignACE)

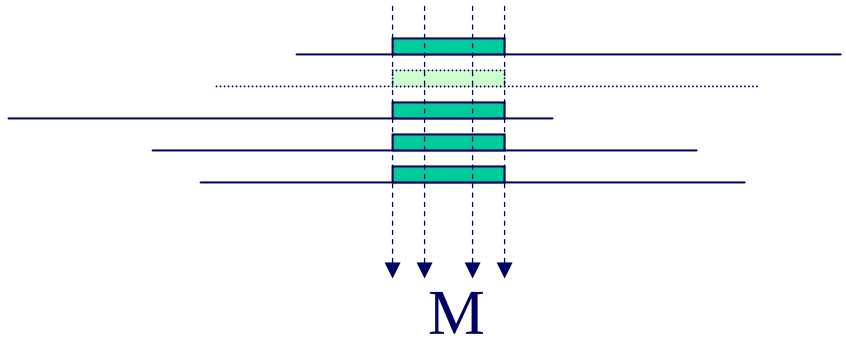
Initialization:

- Select random locations $a_1,\ldots, a_N$ in $x^1, \ldots, x^N$
- For these locations, compute M:

$$M_{kj} = \frac{1}{N}\sum_{i=1}^{N}(x_{a_i+k} = j)$$

- That is, M_{kj} is the number of occurrences of letter j in motif position k, over the total

Gibbs Sampling (AlignACE)



Predictive Update:

- Select a sequence $x = x^i$
- Remove x^i, recompute model:

$$M_{kj} = \frac{1}{(N-1)+B}(\beta_j + \sum_{s=1, s\neq i}^{N}(x_{a_s+k} = j))$$

where β_j are pseudocounts to avoid 0s,
and $B = \Sigma_j \beta_j$

Gibbs Sampling (AlignACE)

Sampling:

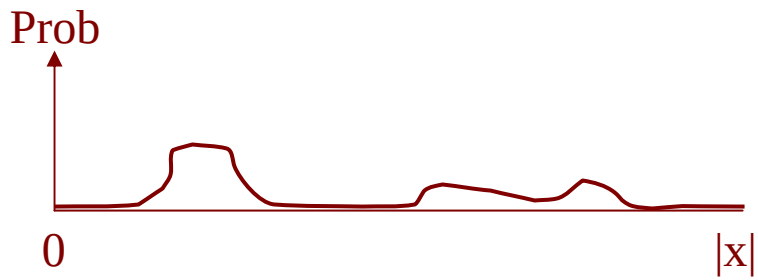
For every K-long word $x_j,\ldots,x_{j+k-1}$ in x:

Q_j = Prob[word | motif] = $M(1,x_j)\times\ldots\times M(k,x_{j+k-1})$

P_i = Prob[word | background] $B(x_j)\times\ldots\times B(x_{j+k-1})$

Let

$$A_j = \frac{Q_j / P_j}{\sum_{j=1}^{|x|-k+1} Q_j / P_j}$$



Sample a random new position a_i according to the probabilities $A_1,\ldots, A_{|x|-k+1}$.

Gibbs Sampling (AlignACE)

Running Gibbs Sampling:

1. Initialize

2. Run until convergence

3. Repeat 1,2 several times, report common motifs

Advantages / Disadvantages

- Very similar to EM

Advantages:

- Easier to implement
- Less dependent on initial parameters
- More versatile, easier to enhance with heuristics

Disadvantages:

- More dependent on all sequences to exhibit the motif
- Less systematic search of initial parameter space

Repeats, and a Better Background Model

- Repeat DNA can be confused as motif
  - Especially low-complexity CACACA… AAAAA, etc.

Solution:

more elaborate background model

0^{th} order: $B = \{ p_A, p_C, p_G, p_T \}$

1^{st} order: $B = \{ P(A|A), P(A|C), \ldots, P(T|T) \}$

…

K^{th} order: $B = \{ P(X \mid b_1 \ldots b_K); X, b_i \in \{A,C,G,T\} \}$

Has been applied to EM and Gibbs (up to 3^{rd} order)

Example Application: Motifs in Yeast

Group:

Tavazoie et al. 1999, G. Church's lab, Harvard

Data:

- Microarrays on 6,220 mRNAs from yeast Affymetrix chips (Cho et al.)
- 15 time points across two cell cycles

Processing of Data

1. Selection of 3,000 genes

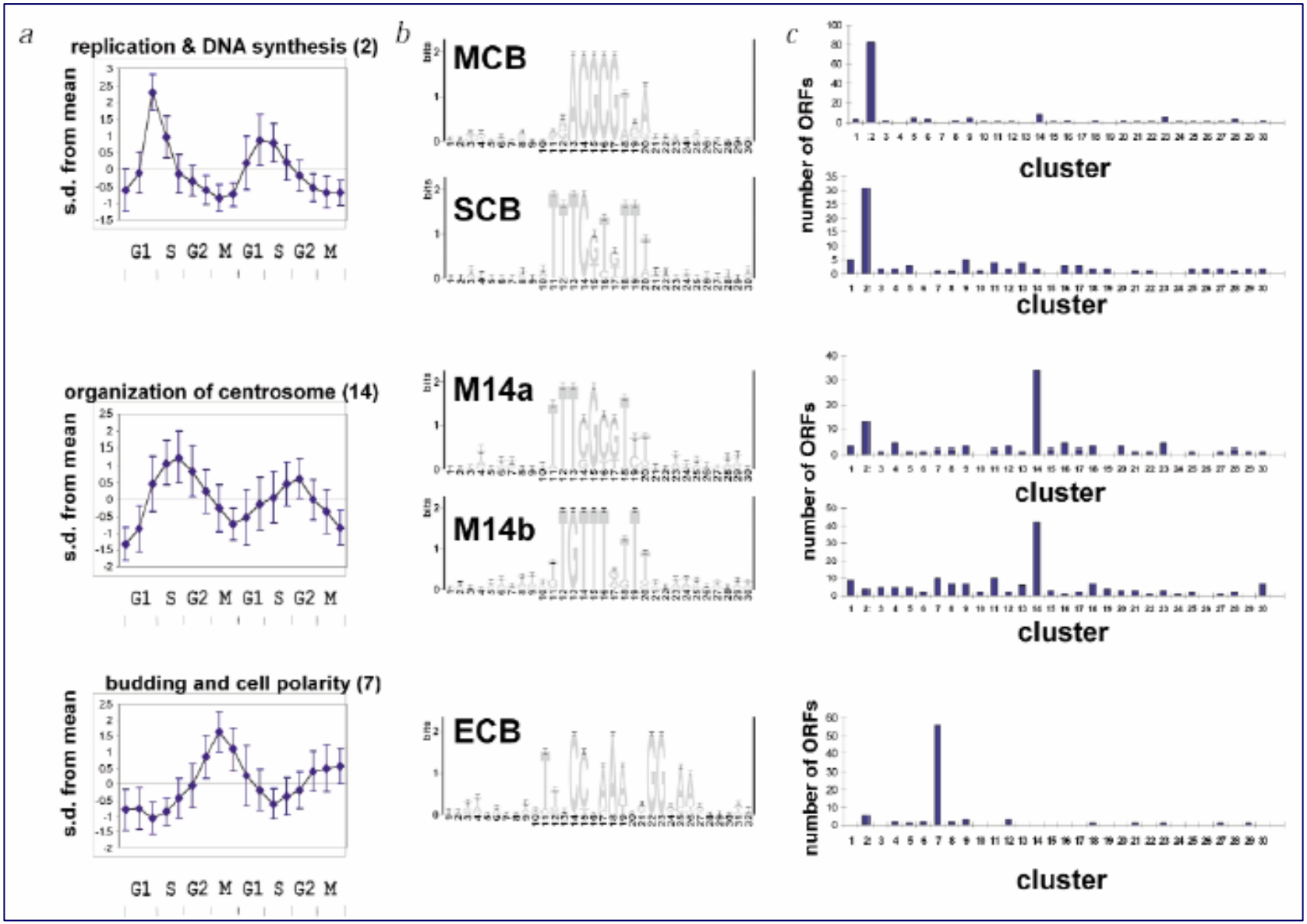
    - Genes with most variable expression were selected

- Clustering according to common expression

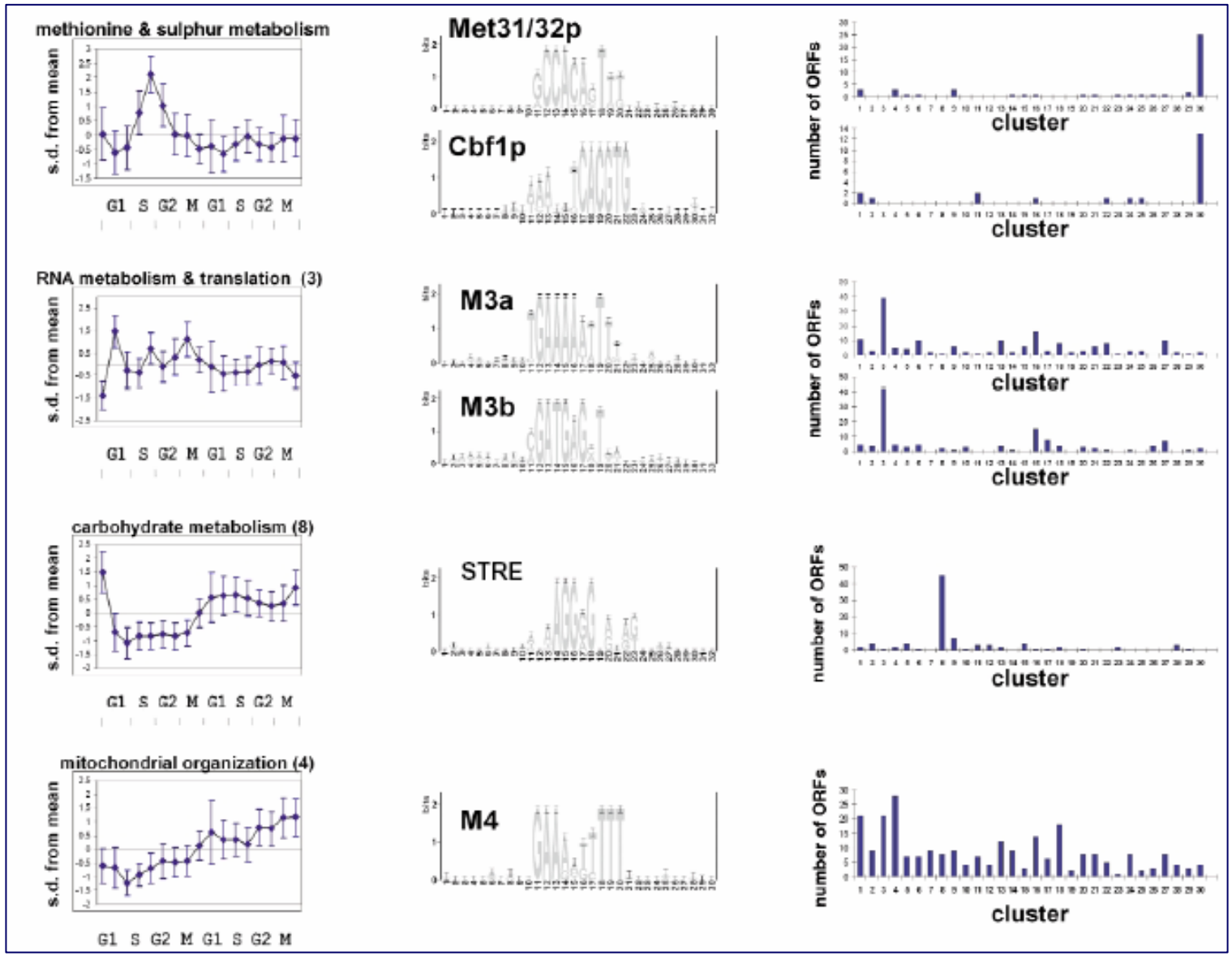
    - K-means clustering
    - 30 clusters, 50-190 genes/cluster
    - Clusters correlate well with known function

1. AlignACE motif finding
    - 600-long upstream regions
    - 50 regions/trial

Motifs in Periodic Clusters



Motifs in Non-periodic Clusters



Overview

- Introduction
  - Bio review: Where do ambiguities come from?
  - Computational formulation of the problem
- Combinatorial solutions
  - Exhaustive search
  - Greedy motif clustering
  - Wordlets and motif refinement
- Probabilistic solutions
  - Expectation maximization
  - MEME extensions
  - Gibbs sampling