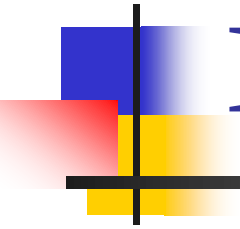
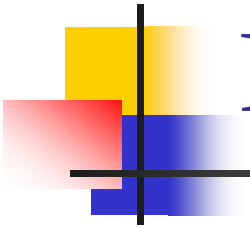


Motif discovery in sequence



Roadmap

- Implanting Patterns in Random Text
- Sequence motifs
- Models for motif representation
- Approaches for motif discovery

Random Sample

atgaccgggatactgataccgtatttggcctaggcgtacacattagataaacgtatgaagtacgttagactcggcgccgccg

acccctatttttgagcagatttagtgacctggaaaaaaaatttgagtacaaaacttttccgaatactgggcataaggtaca

tgagtatccctgggatgacttttgggaacactatagtgctctcccgatttttgaatatgtaggatcattcgccagggtccga

gctgagaattggatgaccttgtaagtgttttccacgcaatcgcgaaccaacgcggacccaaaggcaagaccgataaaggaga

tccctttttgcggtaatgtgccgggaggctggttacgtagggaagccctaacggacttaatggcccacttagtccacttatag

gtcaatcatgttcttgtgaatggatttttaactgagggcatagaccgcttggcgcacccaaattcagtgtgggcgagcgcaa

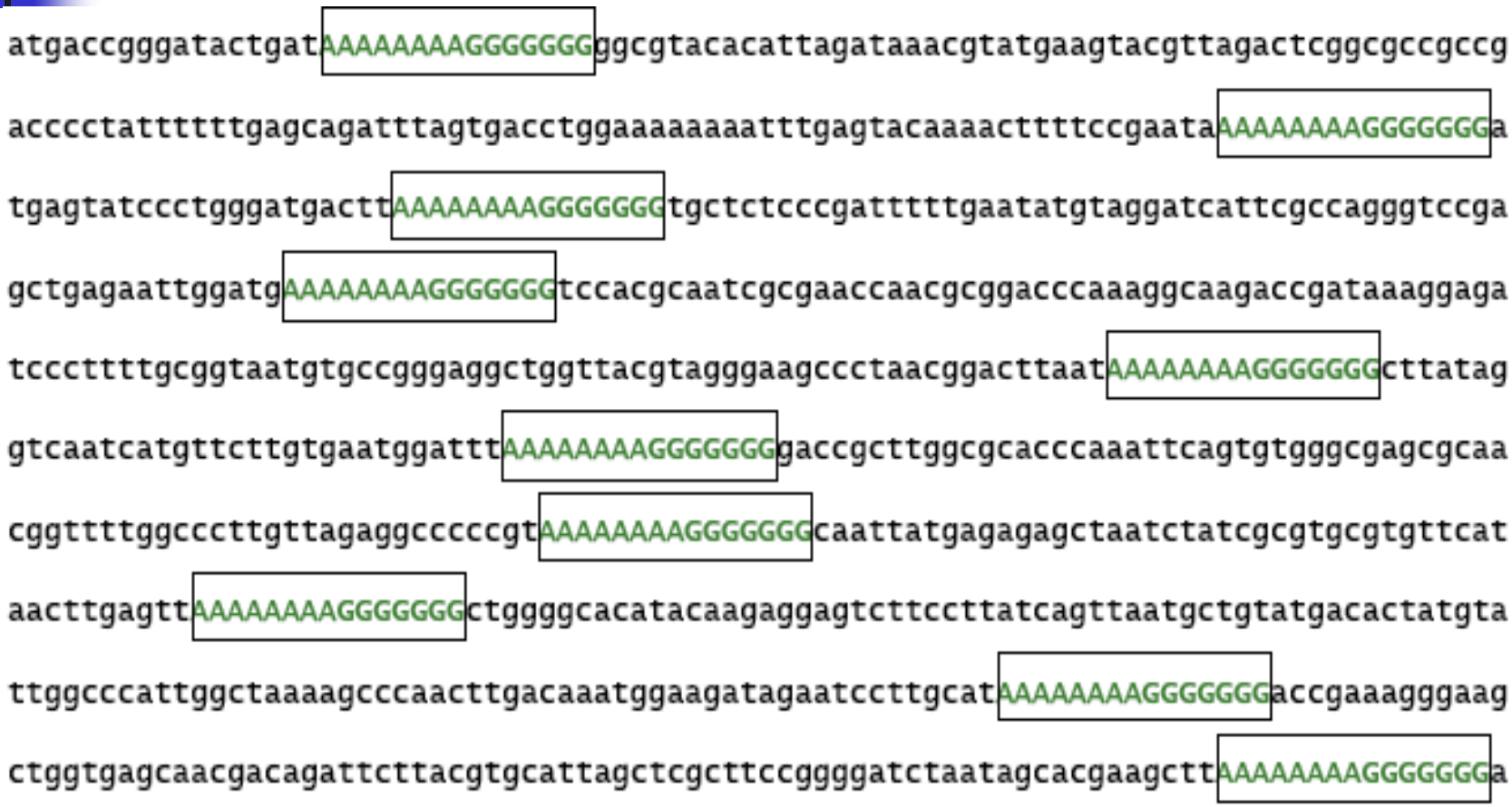
cggttttggcccttgttagaggcccccgtactgatggaaactttcaattatgagagagctaatctatcgcgtgcgtgttcat

aacttgagttggtttcgaaaatgctctggggcacatacaagaggagtcttccttatcagttaatgctgtatgacactatgta

ttggcccattggctaaaagcccaacttgacaaatggaagatagaatccttgcatttcaacgtatgccgaaccgaaagggaag

ctggtgagcaacgacagattcttacgtgcattagctcgcttccggggatctaatagcacgaagcttctgggtactgatagca

Implanting Motif
AAAAAAAGGGGGGG



Where is the Implanted Motif?

atgaccgggatactgatagaagaaaggttgggggcgtacacattagataaacgtatgaagtacgttagactcggcgccgccg

acccctatttttgagcagatttagtgacctggaaaaaaaatttgagtacaaaacttttccgaatacaataaaacggcggga

tgagtatccctgggatgacttaaaataatggagtggtgctctcccgatttttgaatatgtaggatcattcgccagggtccga

gctgagaattggatgcaaaaaaagggattgtccacgcaatcgcgaaccaacgcggacccaaaggcaagaccgataaaggaga

tccctttgcggtaatgtgccgggaggctggttacgtagggaagccctaacggacttaatataataaaggaagggcttatag

gtcaatcatgttcttgtgaatggatttaacaataagggctgggaccgcttggcgcacccaaattcagtgtgggcgagcgcaa

cggttttggcccttgttagaggcccccgtataaacaaggagggccaattatgagagagctaatctatcgcgtgcgtgttcat

aacttgagttaaaaaatagggagccctggggcacatacaagaggagtcttccttatcagttaatgctgtatgacactatgta

ttggcccattggctaaaagcccaacttgacaaatggaagatagaatccttgcatactaaaaaggagcggaccgaaagggaag

ctggtgagcaacgacagattcttacgtgcattagctcgcttccggggatctaatagcacgaagcttactaaaaaggagcgga

Implanting Motif **AAAAAGGGGGGG** with Four Mutations

Where is the Motif???

atgaccgggatactgatagaagaaaggttgggggcgtacacattagataaacgtatgaagtacgttagactcggcgccgccg

acccctatttttgagcagatttagtgacctggaaaaaaaatttgagtacaaaacttttccgaatacaataaaacggcggga

tgagtatccctgggatgacttaaaataatggagtggtgctctcccgatttttgaatatgtaggatcattcgccagggtccga

gctgagaattggatgcaaaaaaagggattgtccacgcaatcgcgaaccaacgcggacccaaaggcaagaccgataaaggaga

tccctttgcggtaatgtgccgggaggctggttacgtagggaagccctaacggacttaatataataaaggaagggcttatag

gtcaatcatgttcttgtgaatggatttaacaataagggctgggaccgcttggcgcacccaaattcagtgtgggcgagcgcaa

cggttttggcccttgttagaggcccccgtataaacaaggagggccaattatgagagagctaatctatcgcgtgcgtgttcat

aacttgagttaaaaaatagggagccctggggcacatacaagaggagtcttccttatcagttaatgctgtatgacactatgta

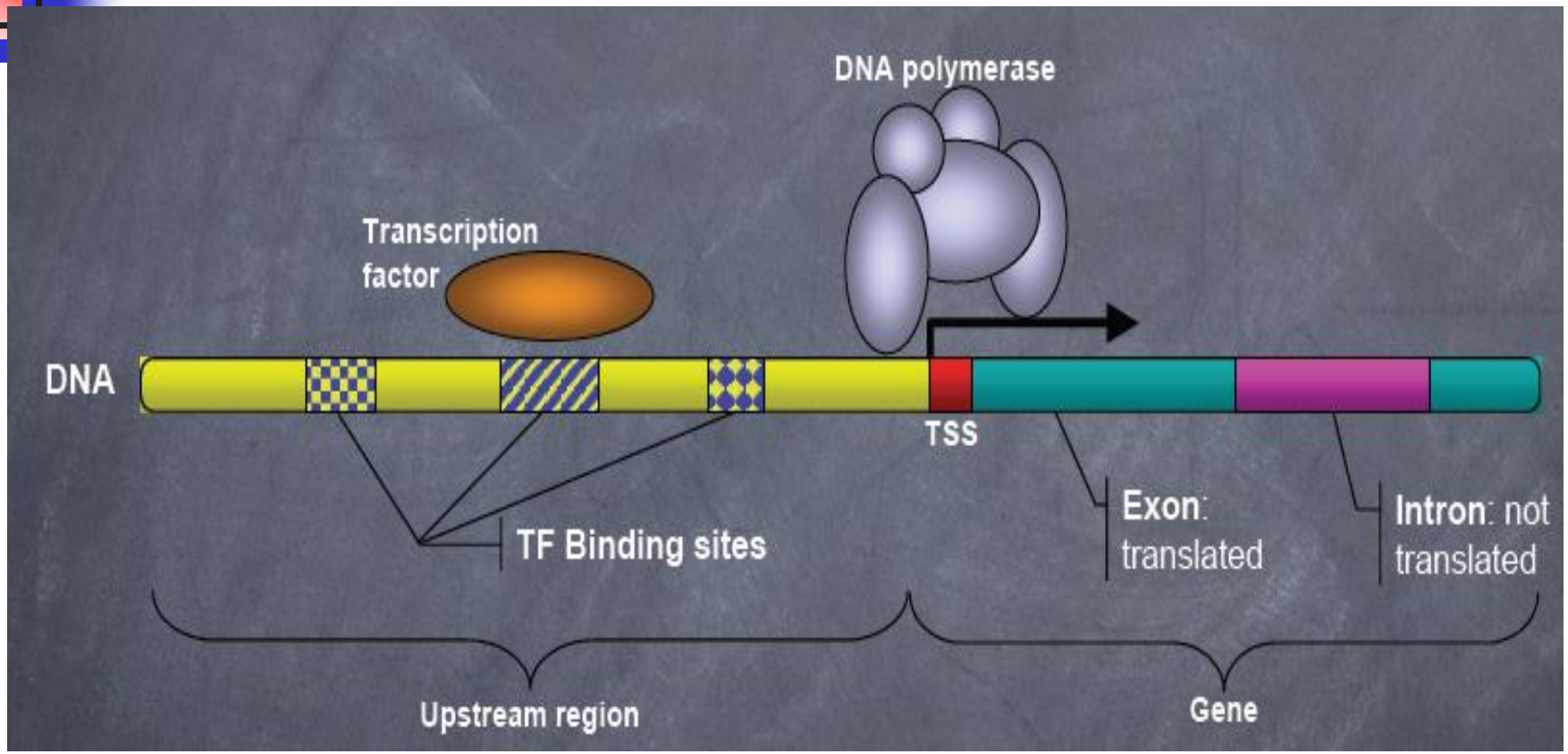
ttggcccattggctaaaagcccaacttgacaaatggaagatagaatccttgcatactaaaaaggagcggaccgaaagggaag

ctggtgagcaacgacagattcttacgtgcattagctcgcttccggggatctaatagcacgaagcttactaaaaaggagcgga

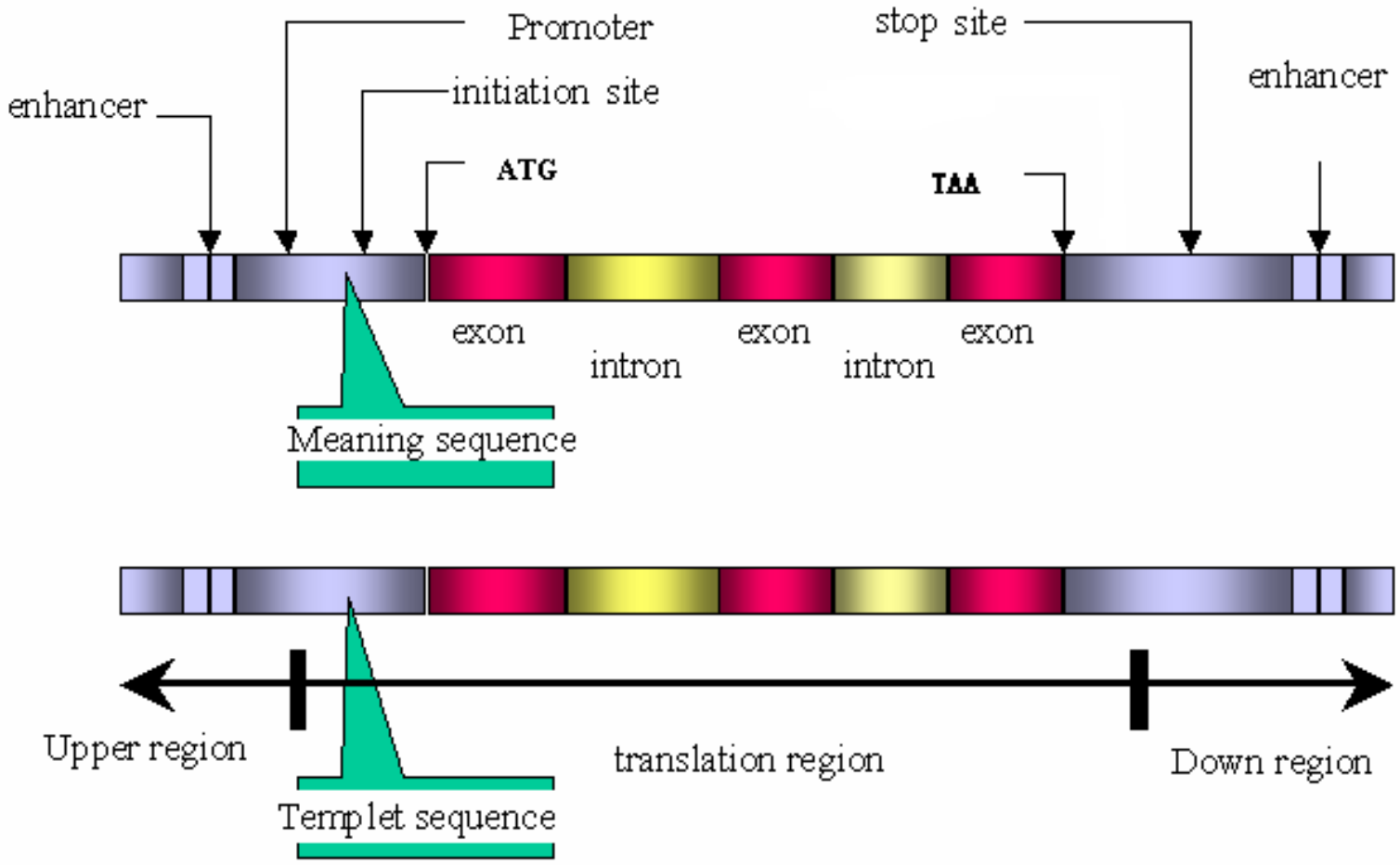
Definition of sequence motif

- A motif is a ***conserved pattern*** found in two or more biological sequences (such as DNA, RNA, or protein sequences), that has specific biological function or structure
- Special regulatory proteins (e.g. ***transcription factors***) and special regulatory sites of DNA sequence (e.g. ***promoters***, ***enhancers***, ***splicing sites***, etc.) .
- Regulatory sites on DNA sequence normally correspond to shared conservative sequence patterns among the regulatory regions of correlated genes. We call these conserved sequence patterns motifs or DNA signals. The actual regulatory DNA sites corresponding to a motif are called the instances of that motif.

TFBS


DNA polymerase
Transcription factor
DNA
TSS
TF Binding sites
Exon: translated
Intron: not translated
Upstream region
Gene

Promoter and Enhancer



Identifying Motifs : Complications

- We do not know the motif sequence
- We do not know where it is located relative to the genes start
- Motifs can differ slightly from one gene to the next
- How to discern区分 it from “random” motifs?

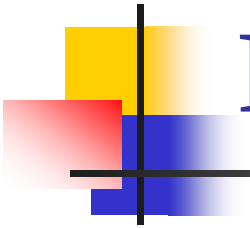
Motif discovery problem

- Given: a data set $S = \{S_1, S_2, \cdots S_m\}$ of *m* DNA sequences (over the alphabet $\Sigma = \{A, C, G, T\}$ with average sequence length *n*, and a parameter *W*, which is the width of the functional related site suspected to be contained in the given data set;

  Objective: find the site instances and a motif model M to represent the conserved site

Approaches for motif discovery

- Approach for consensus motif
  - WINNOWER
  - Random projection approach
- Approach for profile motif
  - Gibbs sampling algorithm
  - MEME-EM algorithm
  - HMM-based approaches



Profiles

- Alignment matrix
- Profile matrix
- Consensus matrix
- Starting position array

$$S = \{a_1, a_2, \cdots a_m\} \qquad 1 \le a_i \le n - w + 1$$

Motifs: Profiles and Consensus

		A	T	C	C	A	G	C	T
		G	G	G	C	A	A	C	T
		A	T	G	G	A	T	C	T
Alignment		A	A	G	C	A	A	C	C
		T	T	G	G	A	A	C	T
		A	T	G	C	C	A	T	T
		A	T	G	G	C	A	C	T

Profile	A	5	1	0	0	5	5	0	0
	T	1	5	0	0	0	1	1	6
	G	1	1	6	3	0	1	0	0
	C	0	0	1	4	2	0	6	1

Consensus								
	A	T	G	C	A	A	C	T

Consensus score

- Consensus score

$$Score(s, DNA) = \sum_{j=1}^{w} M_{P(s)}(j)$$

- P (s) : denote the profile matrix corresponding to start position s
- $M_{P(s)}(j)$ Denote the largest count in column j of P (s)

Exhaustive search

Brute force motif search (DNA, *m*, *n* ,*w*)

bestscore=0

for each $(a_1, a_2, \cdots a_m)$ from (1,1,…,1) to (*n-w+1*, *n-w+1*, …,*n-w+1*)

if Score(s, DNA)> bestscore

bestscore=Score(s,DNA)

bestmotif= $(a_1, a_2, \cdots a_m)$

return bestmotif

Complexity O(wn^m)

Profiles Revisited

- Let **s**=(s_1,...,s_m) be the set of starting positions for *w*-mers in our *m* sequences.
- The substrings corresponding to these starting positions will form:

  - *m* x *w* ***alignment matrix*** and

  - 4 x *w* ***profile matrix**** **P**.

*We make a special note that the profile matrix will be defined in terms of the frequency of letters, and not as the count of letters

Scoring Strings with a Profile

- *Prob*(**a**|**P**) is defined as the probability that an *w*-mer **a** was created by the Profile **P**.
- If **a** is very similar to the consensus string of **P,** then *Prob*(**a**|**P**) will be high
- If **a** is very different, then *Prob*(**a**|**P**) will below.

$$\Pr ob(a|P) = \prod_{i=1}^{m} p_{a_i,i}$$

Scoring Strings with a Profile

- Given a profile: **P =**

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

The probability of the consensus string:

Prob(**aaacct**|**P**) = ???

Scoring Strings with a Profile

- Given a profile: **P =**

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

The probability of the consensus string:

Prob(**aaacct**|**P**) = 1/2 * 7/8 * 3/8 * 5/8 * 3/8* 7/8 = .033646

Scoring Strings with a Profile

- Given a profile: **P =**

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

The probability of the consensus string:
Prob(**aaacct**|**P**) = 1/2 * 7/8 * 3/8 * 5/8 * 3/8 * 7/8 = .033646

Probability of a different string:

Prob(**atacag**|**P**) = 1/2 * 1/8 * 3/8 * 5/8 * 1/8 * 1/8 = .001602

P-Most Probable *w*-mer

- Define the **P**-most probable *w*-mer from a sequence as an *w*-mer in that sequence which has the highest probability of being created from the profile **P**.

P =

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

- Given a sequence = ctataaaccttacatc, find the P-most probable *w*-mer

P-Most Probable *w*-mer

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

Find the *Prob*(**a**|**P**) of every possible 6-mer:

First try: **c t a t a a a c c t t a c a t c**

Second try: **c t a t a a a c c t t a c a t c**

Third try: **c t a t a a a c c t t a c a t c**

-Continue this process to evaluate every possible 6-mer

P-Most Probable *w*-mer

- Compute *prob*(**a**|**P**) for every possible 6-mer:

String, Highlighted in Red	Calculations	*prob*(**a**|**P**)
ctataaaccttacat	1/8 x 1/8 x 3/8 x 0 x 1/8 x 0	0
ctataaaccttacat	1/2 x 7/8 x 0 x 0 x 1/8 x 0	0
ctataaaccttacat	1/2 x 1/8 x 3/8 x 0 x 1/8 x 0	0
ctataaaccttacat	1/8 x 7/8 x 3/8 x 0 x 3/8 x 0	0
ctataaaccttacat	1/2 x 7/8 x 3/8 x 5/8 x 3/8 x 7/8	.0336
ctataaaccttacat	1/2 x 7/8 x 1/2 x 5/8 x 1/4 x 7/8	.0299
ctataaaccttacat	1/2 x 0 x 1/2 x 0 1/4 x 0	0
ctataaaccttacat	1/8 x 0 x 0 x 0 x 0 x 1/8 x 0	0
ctataaaccttacat	1/8 x 1/8 x 0 x 0 x 3/8 x 0	0
ctataaaccttacat	1/8 x 1/8 x 3/8 x 5/8 x 1/8 x 7/8	.0004

P-Most Probable *w*-mer

- P-Most Probable 6-mer in the sequence is aaacct:

String, Highlighted in Red	Calculations	*Prob*(**a**|**P**)
ctataaaccttacat	1/8 x 1/8 x 3/8 x 0 x 1/8 x 0	0
ctataaaccttacat	1/2 x 7/8 x 0 x 0 x 1/8 x 0	0
ctataaaccttacat	1/2 x 1/8 x 3/8 x 0 x 1/8 x 0	0
ctataaaccttacat	1/8 x 7/8 x 3/8 x 0 x 3/8 x 0	0
ctataaaccttacat	**1/2 x 7/8 x 3/8 x 5/8 x 3/8 x 7/8**	**.0336**
ctataaaccttacat	1/2 x 7/8 x 1/2 x 5/8 x 1/4 x 7/8	.0299
ctataaaccttacat	1/2 x 0 x 1/2 x 0 1/4 x 0	0
ctataaaccttacat	1/8 x 0 x 0 x 0 x 0 x 1/8 x 0	0
ctataaaccttacat	1/8 x 1/8 x 0 x 0 x 3/8 x 0	0
ctataaaccttacat	1/8 x 1/8 x 3/8 x 5/8 x 1/8 x 7/8	.0004

P-Most Probable *w*-mer

- **aaacct** is the **P**-most probable 6-mer in:

  ctataaaccttacatc

- because *Prob*(**aaacct**|**P**) = .0336 is greater than the *Prob*(**a**|**P**) of any other 6-mer in the sequence.

P-Most Probable *w*-mers in Many Sequences

Find the **P**-most probable *l*-mer in each of the sequences.

P=

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

ctataaacgttacatc

atagcgattcgactg

cagcccagaaccct

cggtataccttacatc

tgcattcaatagctta

tatcctttccactcac

ctccaaatcctttaca

ggtcatcctttatcct

P-Most Probable *w*-mers in Many Sequences

1	a	a	a	c	g	t
2	a	t	a	g	c	g
3	a	a	c	c	c	t
4	g	a	a	c	c	t
5	a	t	a	g	c	t
6	g	a	c	c	t	g
7	a	t	c	c	t	t
8	t	a	c	c	t	t
A	5/8	5/8	4/8	0	0	0
C	0	0	4/8	6/8	4/8	0
T	1/8	3/8	0	0	3/8	6/8
G	2/8	0	0	2/8	1/8	2/8

ctataaacgttacatc

atagcgattcgactg

cagcccagaaccct

cggtgaaccttacatc

tgcattcaatagctta

tgtcctgtccactcac

ctccaaatcctttaca

ggtctaccttttatcct

P-Most Probable *l*-mers form a new profile

Comparing New and Old Profiles

1	a	a	a	c	g	t
2	a	t	a	g	c	g
3	a	a	c	c	c	t
4	g	a	a	c	c	t
5	a	t	a	g	c	t
6	g	a	c	c	t	g
7	a	t	c	c	t	t
8	t	a	c	c	t	t
A	5/8	5/8	4/8	0	0	0
C	0	0	4/8	6/8	4/8	0
T	1/8	3/8	0	0	3/8	6/8
G	2/8	0	0	2/8	1/8	2/8

A	1/2	7/8	3/8	0	1/8	0
C	1/8	0	1/2	5/8	3/8	0
T	1/8	1/8	0	0	1/4	7/8
G	1/4	0	1/8	3/8	1/4	1/8

Red – frequency increased, Blue – frequency descreased

Greedy Profile Motif Search

- Use P-Most probable *w*-mers to adjust start positions until we reach a "best" profile, this is the motif
  - Select random starting positions
  - Create a profile **P** from the substrings at these starting positions
  - Find the **P**-most probable *w*-mer **a** in each sequence and change the starting position to the starting position of **a**
  - Compute a new profile based on the new starting positions after each iteration and proceed until we cannot increase the score anymore

Greedy Profile Motif Search Algorithm

1. **Greedy Profile Motif Search** *(DNA, m, n, w)*
2. Randomly select starting positions **s**=(s_1,…,s_m) from *DNA*
3. *bestScore* ← *0*
4. **while** Score(**s**, *DNA*) > *bestScore*
5. Form profile **P** from **s**
6. *bestScore* ← Score(**s**, *DNA*)
7. **for** *i* ← *1* **to** *m*
8. Find a **P**-most probable *w*-mer **a** from the i^{th} sequence
9. s_i ← starting position of **a**
10. **return** *bestScore*

Gibbs Sampling

- Greedy Profile Motif Search is probably not the best way to find motifs
- However, we can improve the algorithm by introducing **Gibbs Sampling**, an iterative procedure that discards one *w*-mer after each iteration and replaces it with a new one
- Gibbs Sampling proceeds more slowly and chooses new *w*-mers at random increasing the odds几率 that it will converge to the correct solution

How Gibbs Sampling Works

1) Randomly choose starting positions

   s = (s_1,...,s_m) and form the set of *w*-mers associated with these starting positions.

2) Randomly choose one of the *m* sequences.

3) Create a profile **P** **and the background frequencies Q** from the other *m* -1 sequences.

4) For each position in the removed sequence, calculate the probability that the *w*-mer starting at that position was generated by **P** and **Q**.

5) Choose a new starting position for the removed sequence at random based on the probabilities calculated in step 4.

6) Repeat steps 2-5 until there is no improvement

Gibbs Sampling Algorithm

1. Select a random position in each sequence

Gibbs Sampling Algorithm

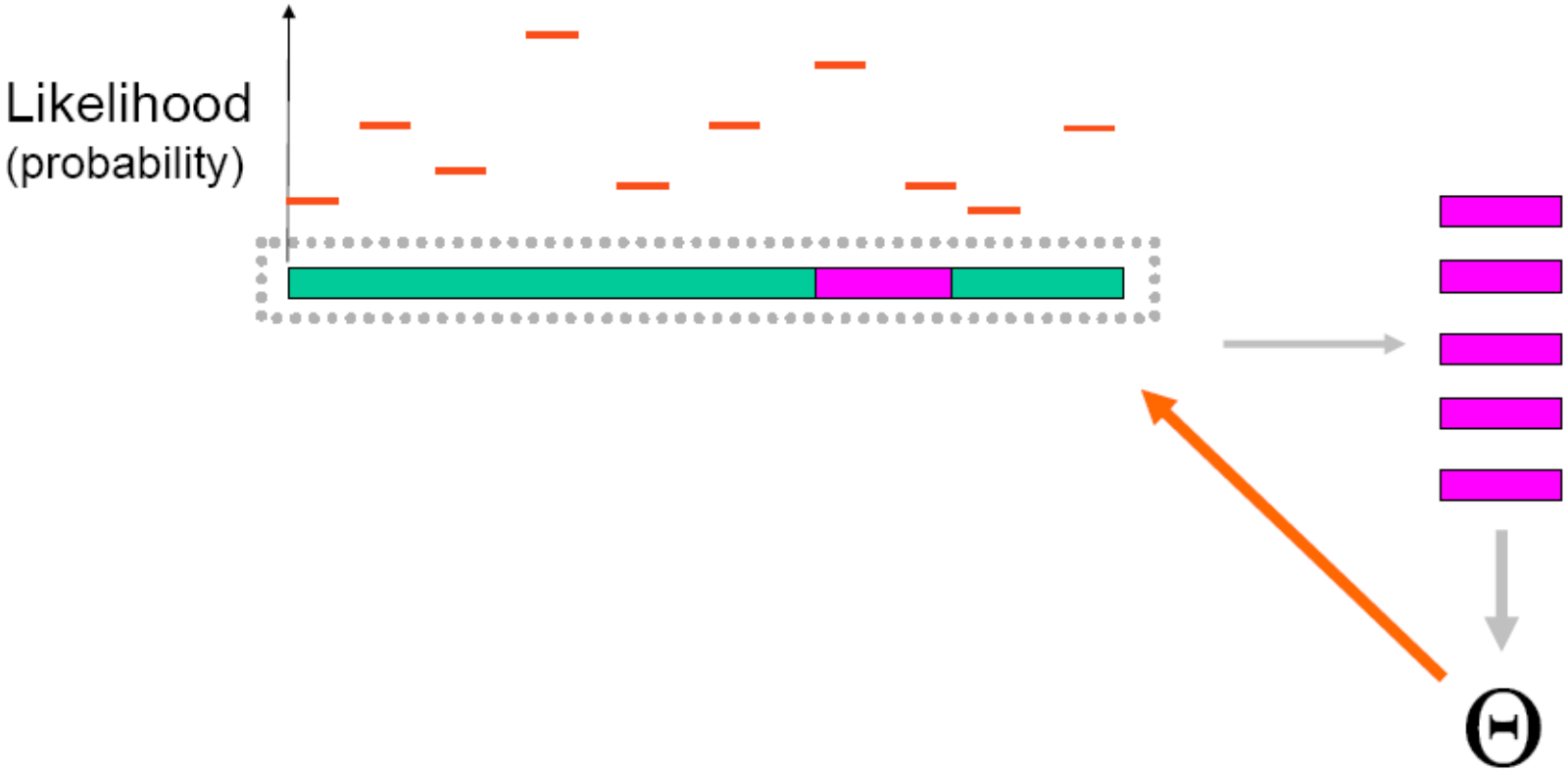
2. Build a weight matrix

Gibbs Sampling Algorithm

3. Select a sequence at random

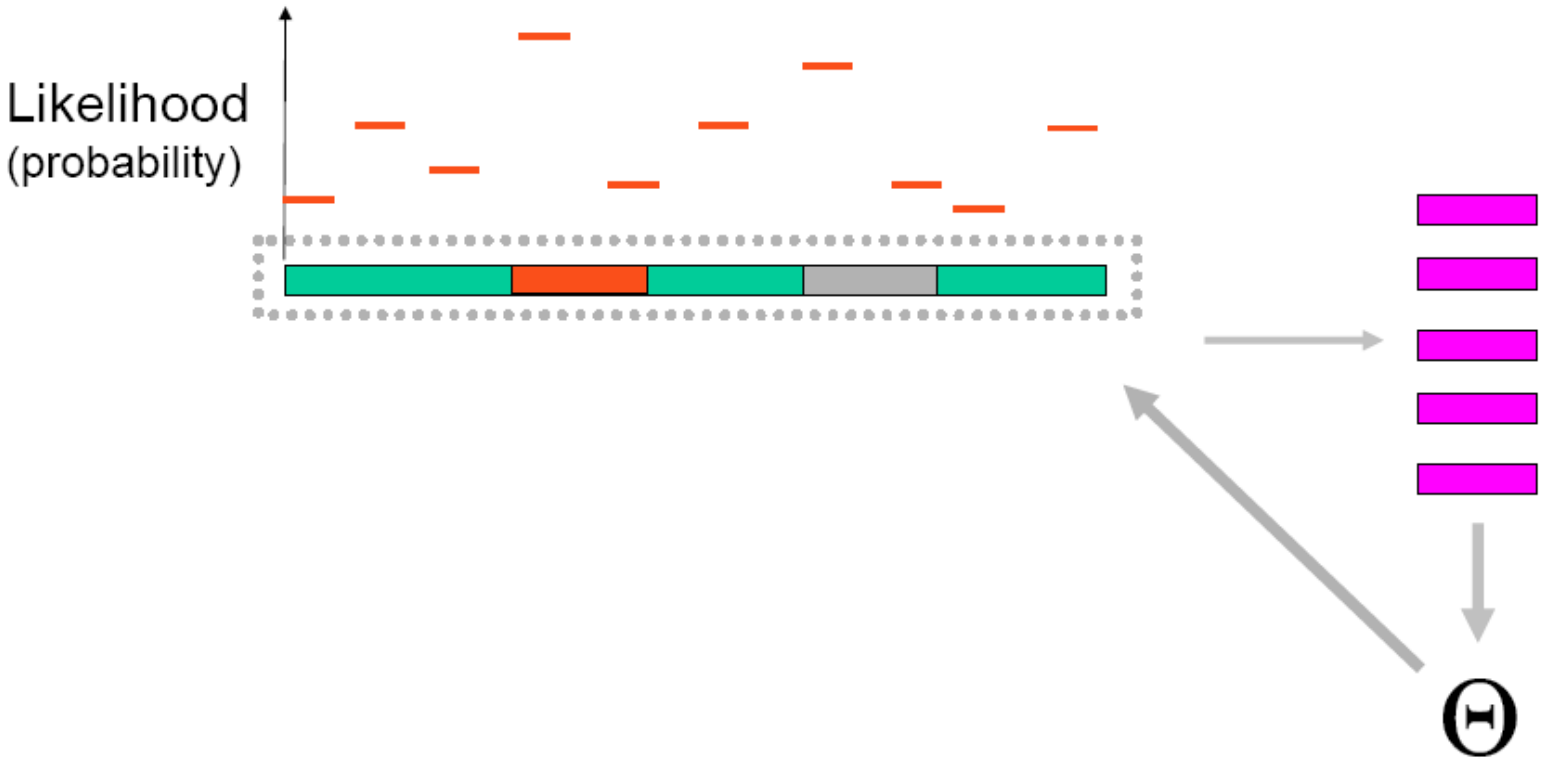
Gibbs Sampling Algorithm

4. Score possible sites in seq using weight matrix



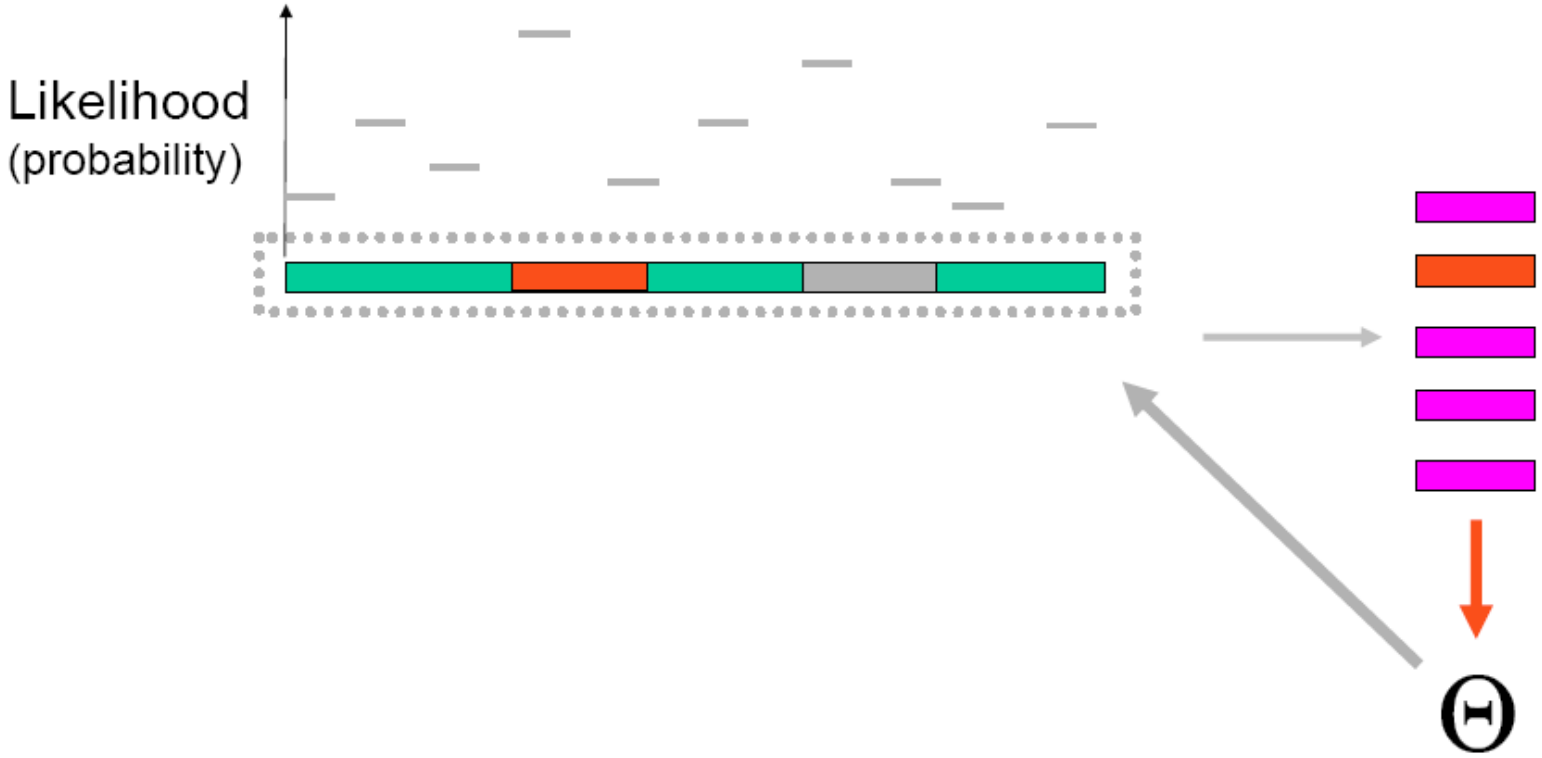
Gibbs Sampling Algorithm

5. Sample a new site proportional to likelihood



Gibbs Sampling Algorithm

6. Update weight matrix



Gibbs Sampling Algorithm

7. Iterate until convergence (no change in sites/Θ)

Gibbs Sampling: an Example

Input:

m = 5 sequences, motif length w= 8

1. GTAAACAATATTTATAGC
2. AAAATTTACCTCGCAAGG
3. CCGTACTGTCAAGCGTGG
4. TGAGTAAACGACGTCCCA
5. TACTTAACACCCTGTCAA

Gibbs Sampling: an Example

1) Randomly choose starting positions, $\boldsymbol{s}=(s_1,s_2,s_3,s_4,s_5)$ in the 5 sequences:

$s_1=7$	GTAAACAATATTTATAGC
$s_2=11$	AAAATTTACCTTAGAAGG
$s_3=9$	CCGTACTGTCAAGCGTGG
$s_4=4$	TGAGTAAACGACGTCCCA
$s_5=1$	TACTTAACACCCTGTCAA

Gibbs Sampling: an Example

2) Choose one of the sequences at random:

Sequence 2: AAAATTTACCTTAGAAGG

$s_1=7$	GTAAACAATATTTATAGC
$s_2=11$	AAAATTTACCTTAGAAGG
$s_3=9$	CCGTACTGTCAAGCGTGG
$s_4=4$	TGAGTAAACGACGTCCCA
$s_5=1$	TACTTAACACCCTGTCAA

Gibbs Sampling: an Example

3) a) Create profile ***P*** from *w*-mers in remaining 4 sequences:

1	A	A	T	A	T	T	T	A
3	T	C	A	A	G	C	G	T
4	G	T	A	A	A	C	G	A
5	T	A	C	T	T	A	A	C
A	1/4	2/4	2/4	3/4	1/4	1/4	1/4	2/4
C	0	1/4	1/4	0	0	2/4	0	1/4
T	2/4	1/4	1/4	1/4	2/4	1/4	1/4	1/4
G	1/4	0	0	0	1/4	0	3/4	0
Consensus String	T	A	A	A	T	C	G	A

Gibbs Sampling: an Example

3) b) Create profile ***Q*** from the remaining 4 sequences, not containing the pattern

s_1=*7* GTAAACAATATTTATAGC

s_2=11 **AAAATTTACCTTAGAAGG**

s_3=*9* CCGTACTGTCAAGCGTGG

s_4=*4* TGAGTAAACGACGTCCCA

s_5=*1* TACTTAACACCCTGTCAA

Q=(10/40, 13/40,9/40,8/40)=(0.25,0.33,0.23.0.2)

Gibbs Sampling: an Example

4) a) Calculate the *prob*(***a***|***P***) for every possible 8-mer in the removed sequence:

Strings Highlighted in Red	*prob*(**a**|**P**)
AAAATTTACCTTAGAAGG	.000732
AAAATTTACCTTAGAAGG	.000122
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	.000183
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	0
AAAATTTACCTTAGAAGG	0

Gibbs Sampling: an Example

4) b) Calculate the *prob*(***a***|***Q***) for every possible 8-mer in the removed sequence:

Strings Highlighted in Red	*prob*(**a**|**Q**)
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	
AAAATTTACCTTAGAAGG	

Gibbs Sampling: an Example

5) Create a distribution of probabilities of *w*-mers *prob*(***a***|***Q***), and randomly select a new starting position based on this distribution.

a) To create this distribution, divide each probability *prob*(***a***|***P***) by the probability *prob*(***a***|***Q***) :

Turning Probabilities into Ratios

b) Define probabilities of starting positions according to computed ratios

c) Select the start position according to computed ratios:

P(selecting starting position 1): .706

P(selecting starting position 2): .118

P(selecting starting position 8): .176

Gibbs Sampling: an Example

- Assume we select the substring with the highest probability – then we are left with the following new substrings and starting positions.

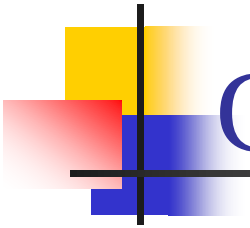
$s_1=7$ GTAAACAATATTTATAGC

$s_2=1$ AAAATTTACCTCGCAAGG

$s_3=9$ CCGTACTGTCAAGCGTGG

$s_4=4$ TGAGTAATCGACGTCCCA

$s_5=1$ TACTTCACACCCTGTCAA



Gibbs Sampling: an Example

6) We iterate the procedure again with the above starting positions until we cannot improve the ***score*** any more.