DNA1: Last week's take-home lessons

Types of mutants
Mutation, drift, selection
    Binomial for each
Association studies χ^2 statistic
Linked & causative alleles
Alleles, Haplotypes, genotypes
Computing the first genome,
    the second ...
New technologies
Random and systematic errors

DNA2: Today's story and goals

Motivation and connection to DNA1

Comparing types of alignments & algorithms

Dynamic programming

Multi-sequence alignment

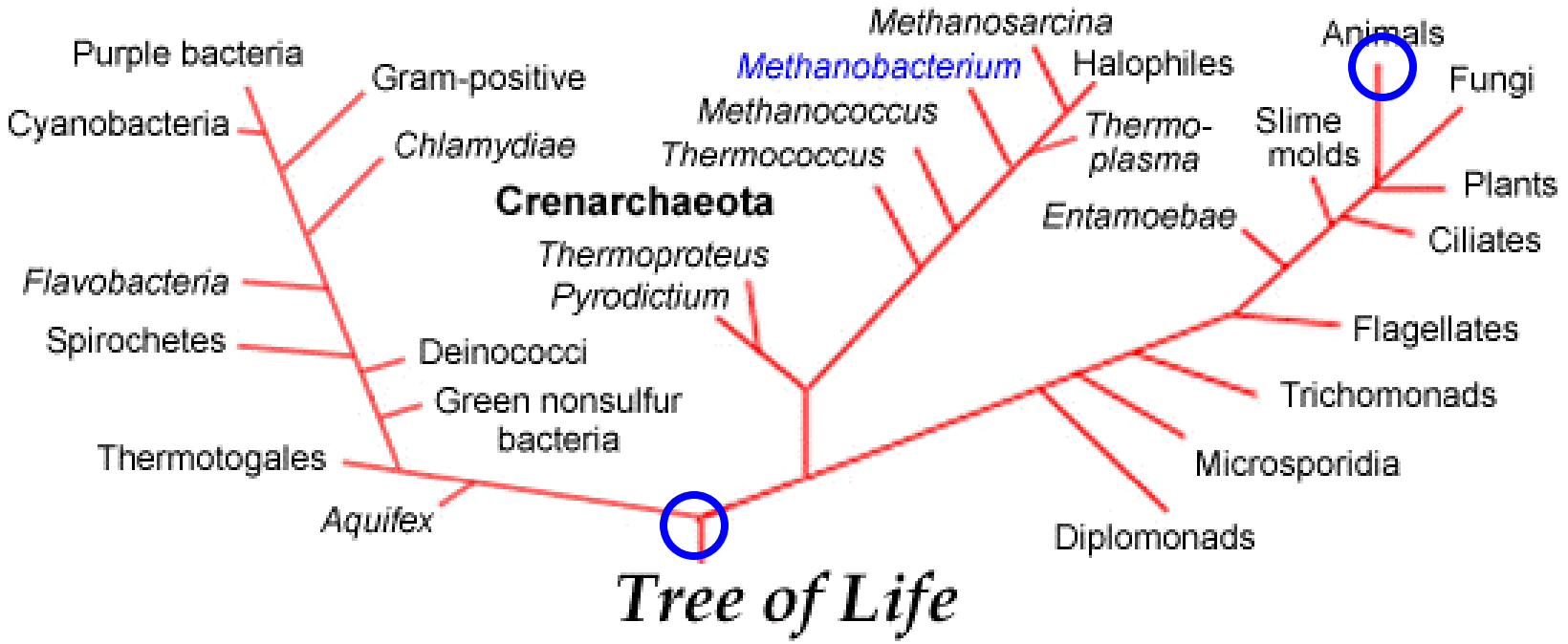
Space-time-accuracy tradeoffs

Finding genes -- motif profiles

Hidden Markov Model for CpG Islands

DNA 2

DNA1: the last 5000 generations



Intro2: Common & simple

Figure **(http://216.190.101.28/GOLD/)**

Applications of Dynamic Programming

- To sequence analysis
  - Shotgun sequence assembly
  - Multiple alignments
  - Dispersed & tandem repeats
  - Bird song alignments
  - Gene Expression time-warping
- Through HMMs
  - RNA gene search & structure prediction
  - Distant protein homologies
  - Speech recognition

Alignments & Scores

Global (e.g. haplotype)

```
ACCACACA
::xx::x:
ACACCATA
```

Score= 5(+1) + 3(-1) = 2

Local (motif)

```
ACCACACA
::::
ACACCATA
```

Score= 4(+1) = 4

Suffix (shotgun assembly)

```
ACCACACA
:::
ACACCATA
```

Score= 3(+1) =3

Increasingly complex (accurate) searches

Exact (StringSearch)	CGCG
Regular expression (PrositeSearch)	CGN{0-9}CG = CGAACG
Substitution matrix (BlastN)	CGCG ~= CACG
Profile matrix (PSI-blast)	CGc(g/a) ~= CACG
Gaps (Gap-Blast)	CGCG ~= CGAACG
Dynamic Programming (NW, SM)	CGCG ~= CAGACG

Hidden Markov Models (HMMER)

"Hardness" of (multi-) sequence alignment

Align 2 sequences of length N allowing gaps.

```
ACCAC-ACA        ACCACACA
::x::x:x:        :xxxxxx:
AC-ACCATA   ,    A-----CACCATA   , etc.
```

2N gap positions, gap lengths of 0 to N each:
A naïve algorithm might scale by $O(N^{2N})$.
For $N= 3x10^9$ this is rather large.

Now, what about k>2 sequences?
or rearrangements other than gaps?

Testing search & classification algorithms

Separate Training set and Testing sets
Need databases of non-redundant sets.
Need evaluation criteria (programs)
Sensitivity and Specificity (false negatives & positives)

sensitivity (true_predicted/true)
specificity (true_predicted/all_predicted)

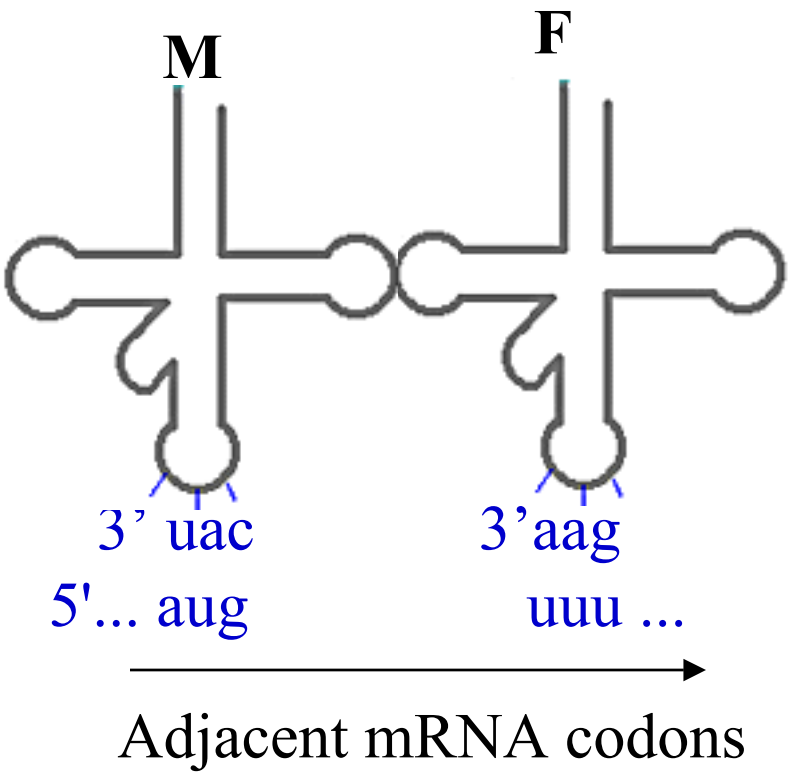
Where do training sets come from?
More expensive experiments: crystallography, genetics, biochemistry

Comparisons of homology scores

Pearson WR Protein Sci 1995 Jun;4(6):1145-60
Comparison of methods for searching protein sequence databases. Methods Enzymol 1996;266:227-58
Effective protein sequence comparison.

Algorithm: FASTA, Blastp, Blitz
Substitution matrix:PAM120, PAM250, BLOSUM50, BLOSUM62
Database: PIR, SWISS-PROT, GenPept

Switch to protein searches when possible



x=	u	c	a	g	
Uxu	F	S	Y	C	
uxc					
uxa	L		-	-	TER
uxg			-	W	
Cxu		P	H	R	
cxc					
cxa			Q		
cxg					
axu	I	T	N	S	
axc					C-S
axa			K	R	
axg	M				NH+
gxu	V	A	D	G	
gxc					O-
gxa			E		
gxg					H:D/A

A Multiple Alignment of Immunoglobulins

```
VTISCTGSSSNIGAG-NHVKWYQQLPG
VTISCTGTSSNIGS--ITVNWYQQLPG
LRLSCSSSGFIFSS--YAMYWVRQAPG
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
ATLVCLISDFYPGA--VTVAWKADS--
AALGCLVKDYFPEP--VTVSWNSG---
VSLTCLVKGFYPSD--IAVEWESNG--
```

Scoring matrix based on large set of distantly related blocks: Blosum62

10	1	6	6	4	8	2	6	6	**9**	2	4	4	4	5	6	6	7	1	3	%
A	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**K**	**L**	**M**	**N**	**P**	**Q**	**R**	**S**	**T**	**V**	**W**	**Y**	
8	0	-4	-2	-4	0	-4	-2	-2	-2	-2	-4	-2	-2	-2	2	0	0	-6	-4	**A**
	18	-6	-8	-4	-6	-6	-2	-6	-2	-2	-6	-6	-6	-6	-2	-2	-2	-4	-4	**C**
		12	**4**	-6	-2	-2	-6	-2	-8	-6	2	-2	0	-4	0	-2	-6	-8	-6	**D**
			10	-6	-4	0	-6	2	-6	-4	0	-2	**4**	0	0	-2	-4	-6	-4	**E**
				12	-6	-2	0	-6	0	0	-6	-8	-6	-6	-4	-4	-2	2	**6**	**F**
					12	-4	-8	-4	-8	-6	0	-4	-4	-4	0	-4	-6	-4	-6	**G**
						16	-6	-2	-6	-4	2	-4	0	0	-2	-4	-6	-4	4	**H**
							8	-6	4	2	-6	-6	-6	-6	-4	-2	6	-6	-2	**I**
								10	-4	-2	0	-2	2	**4**	0	-2	-4	-6	-4	**K**
									8	4	-6	-6	-4	-4	-4	-2	2	-4	-2	**L**
										10	-4	-4	0	-2	-2	-2	2	-2	-2	**M**
											12	-4	0	0	2	0	-6	-8	-4	**N**
												14	-2	-4	-2	-2	-4	-8	-6	**P**
													10	2	0	-2	-4	-4	-2	**Q**
														10	-2	-2	-6	-6	-4	**R**
															8	2	-4	-6	-4	**S**
																10	0	-4	-4	**T**
																	8	-6	-2	**V**
																		22	**4**	**W**
																			14	**Y**

Scoring Functions and Alignments

- Scoring function:

  ω(match) = +1;

  ω(mismatch) = -1;

  ω(indel) = -2;

  ω(other) = 0.

  } substitution matrix

- Alignment score: sum of columns.
- Optimal alignment: maximum score.

Calculating Alignment Scores

```
(1)  ATGA
     :xx:
     ACTA
```

```
(2)  A-TGA
     : : :
     ACT-A
```

$$(1)\textbf{Score} = \omega\begin{pmatrix} A \\ A \end{pmatrix} + \omega\begin{pmatrix} T \\ C \end{pmatrix} + \omega\begin{pmatrix} G \\ T \end{pmatrix} + \omega\begin{pmatrix} A \\ A \end{pmatrix} = 1 - 1 - 1 + 1 = 0.$$

$$(2)\textbf{Score} = \omega\begin{pmatrix} A \\ A \end{pmatrix} + \omega\begin{pmatrix} - \\ C \end{pmatrix} + \omega\begin{pmatrix} T \\ T \end{pmatrix} + \omega\begin{pmatrix} G \\ - \end{pmatrix} + \omega\begin{pmatrix} A \\ A \end{pmatrix} = 1 - 2 + 1 - 2 + 1 = -1.$$

$$\textbf{if } \omega(\textbf{indel}) = -1, \ \textbf{Score} = 1 - 1 + 1 - 1 + 1 = +1.$$

DNA2: Today's story and goals

Motivation and connection to DNA1

Comparing types of alignments & algorithms

Dynamic programming

Multi-sequence alignment

Space-time-accuracy tradeoffs

Finding genes -- motif profiles

Hidden Markov Model for CpG Islands

What is dynamic programming?

A dynamic programming algorithm solves every subsubproblem just once and then saves its answer in a table, avoiding the work of recomputing the answer every time the subsubproblem is encountered.

-- Cormen *et al*. "Introduction to Algorithms", The MIT Press.

Recursion of Optimal Global Alignments

$s\begin{pmatrix} u \\ v \end{pmatrix}$: optimal global alignment score of u and v.

$$s\begin{pmatrix} \mathrm{ATG}A \\ \mathrm{ACT}A \end{pmatrix} = \max\begin{cases} s\begin{pmatrix} \mathrm{ATG}A \\ \mathrm{ACT} \end{pmatrix} + \omega\begin{pmatrix} - \\ A \end{pmatrix}; \\ s\begin{pmatrix} \mathrm{ATG} \\ \mathrm{ACT} \end{pmatrix} + \omega\begin{pmatrix} A \\ A \end{pmatrix}; \\ s\begin{pmatrix} \mathrm{ATG} \\ \mathrm{ACT}A \end{pmatrix} + \omega\begin{pmatrix} A \\ - \end{pmatrix}. \end{cases}$$

Recursion of Optimal Local Alignments

$s\begin{pmatrix} u \\ v \end{pmatrix}$: optimal local alignment score of u and v.

$$s\begin{pmatrix} u_1 u_2 \ldots u_i \\ v_1 v_2 \ldots v_j \end{pmatrix} = \max \begin{cases} s\begin{pmatrix} u_1 u_2 \ldots u_i \\ v_1 v_2 \ldots v_{j-1} \end{pmatrix} + \omega\begin{pmatrix} - \\ v_j \end{pmatrix}; \\ s\begin{pmatrix} u_1 u_2 \ldots u_{i-1} \\ v_1 v_2 \ldots v_{j-1} \end{pmatrix} + \omega\begin{pmatrix} u_i \\ v_j \end{pmatrix}; \\ s\begin{pmatrix} u_1 u_2 \ldots u_{i-1} \\ v_1 v_2 \ldots v_j \end{pmatrix} + \omega\begin{pmatrix} u_i \\ - \end{pmatrix}; \\ 0. \end{cases}$$

Computing Row-by-Row

		A	C	T	A
	0	min	min	min	min
A	min	1	-1	-3	-5
T	min				
G	min				
A	min				

		A	C	T	A
	0	min	min	min	min
A	min	1	-1	-3	-5
T	min	-1	0	0	-2
G	min				
A	min				

		A	C	T	A
	0	min	min	min	min
A	min	1	-1	-3	-5
T	min	-1	0	0	-2
G	min	-3	-2	-1	-1
A	min				

		A	C	T	A
	0	min	min	min	min
A	min	1	-1	-3	-5
T	min	-1	0	0	-2
G	min	-3	-2	-1	-1
A	min	-5	-4	-3	0

$\min = -10^{99}$

Traceback Optimal Global Alignment

		A	C	T	A
	0	min	min	min	min
A	min	1	-1	-3	-5
T	min	-1	0	0	-2
G	min	-3	-2	-1	-1
A	min	-5	-4	-3	0

$$\begin{pmatrix} A & G & T & A \\ \vdots & \times & \times & \vdots \\ A & T & C & A \end{pmatrix}$$

Local and Global Alignments

		A	C	C	A	C	A	C	A
	0	0	0	0	0	0	0	0	0
A	0	1	0	0	1	0	1	0	1
C	0	0	2	1	0	2	0	2	0
A	0	1	0	1	2	0	3	1	3
C	0	0	2	1	0	3	1	4	2
C	0	0	0	3	2	1	2	2	3
A	0	1	0	1	4	2	2	1	3
T	0	0	0	0	2	3	1	1	1
A	0	1	0	0	1	1	4	2	2

		A	C	C	A	C	A	C	A
	0	m	m	m	m	m	m	m	m
A	m	1	-1	-3	-5	-7	-9	-11	-13
C	m	-1	2	0	-2	-4	-6	-8	-10
A	m	-3	0	1	1	-1	-3	-5	-7
C	m	-5	-2	1	0	2	0	-2	-4
C	m	-7	-4	-1	0	1	1	1	-1
A	m	-9	-6	-3	0	-1	2	0	2
T	m	-11	-8	-5	-2	-1	0	1	0
A	m	-13	-10	-7	-4	-1	0	-1	2

Time and Space Complexity of Computing Alignments

For two sequences $u=u_1u_2\ldots u_n$ and $v=v_1v_2\ldots v_m$, finding the optimal alignment takes $O(mn)$ time and $O(mn)$ space.

An $O(1)$-time operation: one comparison, three multiplication steps, computing an entry in the alignment table…

An $O(1)$-space memory: one byte, a data structure of two floating points, an entry in the alignment table…

Time and Space Problems

- Comparing two one-megabase genomes.
- Space:
  - An entry: 4 bytes;
  - Table: 4 * 10^6 * 10^6 = 4 G bytes memory.
- Time:
  - 1000 MHz CPU: 1M entries/second;
  - 10^12 entries: 1M seconds = 10 days.

Time & Space Improvement for w-band Global Alignments

- Two sequences differ by at most w bps (w<<n).
- w-band algorithm: O(wn) time and space.
- Example: w=3.

		A	C	C	A	C	A	C	A
	0	m	m	m					
A	m	1	-1	-3	-5				
C	m	-1	2	0	-2	-4			
A	m	-3	0	1	1	-1	-3		
C		-5	-2	1	0	2	0	-2	
C			-4	-1	0	1	1	1	-1
A				-3	0	-1	2	0	2
T					-2	-1	0	1	0
A						-1	0	-1	2

Summary

Dynamic programming

Statistical interpretation of alignments

Computing optimal global alignment

Computing optimal local alignment

Time and space complexity

Improvement of time and space

Scoring functions

DNA2: Today's story and goals

Motivation and connection to DNA1

Comparing types of alignments & algorithms

Dynamic programming

Multi-sequence alignment

Space-time-accuracy tradeoffs

Finding genes -- motif profiles

Hidden Markov Model for CpG Islands

A Multiple Alignment of Immunoglobulins

```
VTISCTGSSSNIGAG-NHVKWYQQLPG
VTISCTGTSSNIGS--ITVNWYQQLPG
LRLSCSSSGFIFSS--YAMYWVRQAPG
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
ATLVCLISDFYPGA--VTVAWKADS--
AALGCLVKDYFPEP--VTVSWNSG---
VSLTCLVKGFYPSD--IAVEWESNG--
```

A multiple alignment <=> Dynamic programming on a hyperlattice

See G. Fullen, 1996.

Multiple Alignment vs Pairwise Alignment

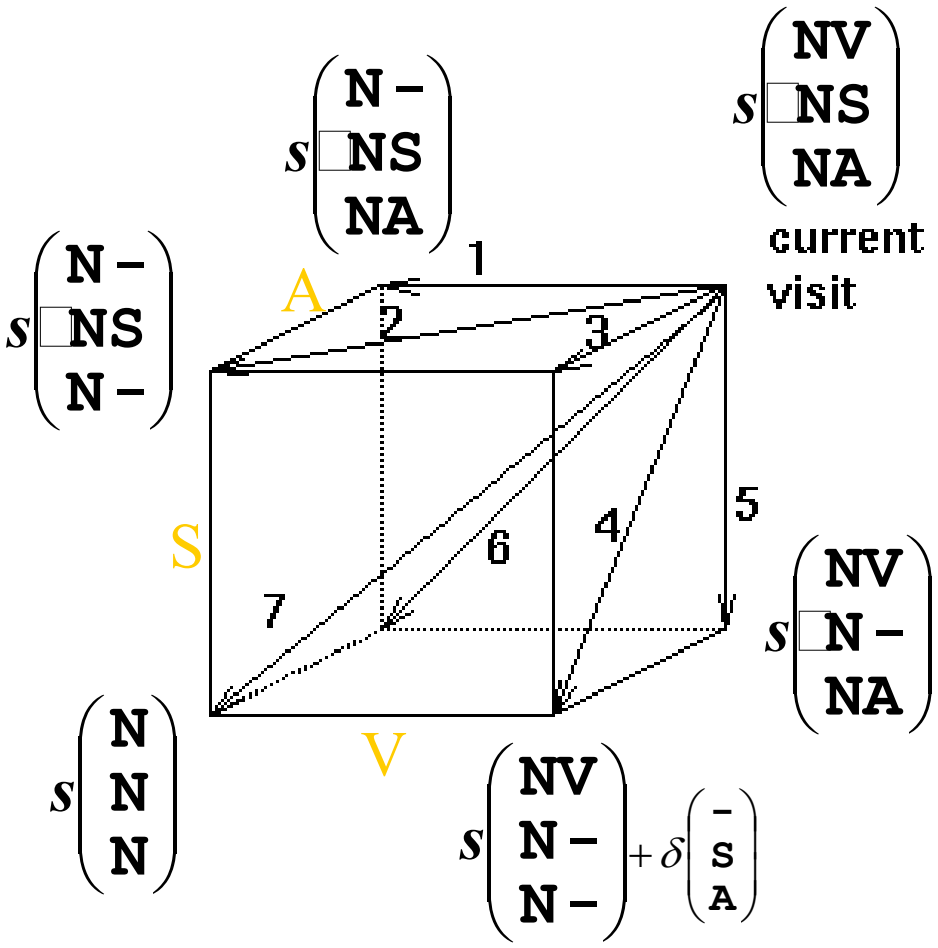
```
AT
AT
AT
A-
-T
AT
AT
```

Optimal Multiple Alignment

```
A-
-T
```

Non-Optimal Pairwise Alignment

Computing a Node on Hyperlattice



$k=3 \qquad 2^k - 1 = 7$

$$s\begin{pmatrix} \mathbf{NV} \\ \mathbf{NS} \\ \mathbf{NA} \end{pmatrix} = \max \left\{ \begin{array}{l} s\begin{pmatrix} \mathbf{N} \\ \mathbf{N} \\ \mathbf{N} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{V} \\ \mathbf{S} \\ \mathbf{A} \end{pmatrix} \\ s\begin{pmatrix} \mathbf{NV} \\ \mathbf{N-} \\ \mathbf{N-} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{-} \\ \mathbf{S} \\ \mathbf{A} \end{pmatrix} \\ s\begin{pmatrix} \mathbf{N-} \\ \mathbf{NS} \\ \mathbf{N-} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{V} \\ \mathbf{-} \\ \mathbf{A} \end{pmatrix} \\ s\begin{pmatrix} \mathbf{N-} \\ \mathbf{N-} \\ \mathbf{NA} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{V} \\ \mathbf{S} \\ \mathbf{-} \end{pmatrix} \\ s\begin{pmatrix} \mathbf{N-} \\ \mathbf{NS} \\ \mathbf{NA} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{V} \\ \mathbf{-} \\ \mathbf{-} \end{pmatrix} \\ s\begin{pmatrix} \mathbf{NV} \\ \mathbf{N-} \\ \mathbf{NA} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{-} \\ \mathbf{S} \\ \mathbf{-} \end{pmatrix} \\ s\begin{pmatrix} \mathbf{NV} \\ \mathbf{NS} \\ \mathbf{N-} \end{pmatrix} + \delta\begin{pmatrix} \mathbf{-} \\ \mathbf{-} \\ \mathbf{A} \end{pmatrix} \end{array} \right.$$

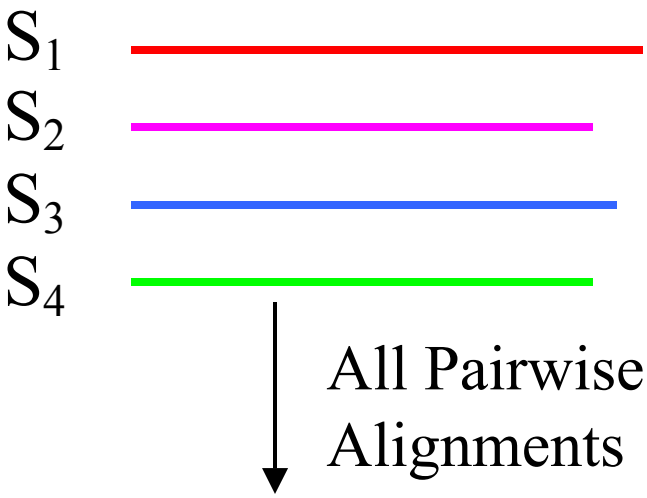
Challenges of Optimal Multiple Alignments

- Space complexity (hyperlattice size): $O(n^k)$ for k sequences each n long.
- Computing a hyperlattice node: $O(2^k)$.
- Time complexity: $O(2^k n^k)$.
- Find the optimal solution is exponential in k (non-polynomial, NP-hard).

Methods and Heuristics for Optimal Multiple Alignments

- Optimal: dynamic programming
  - Pruning the hyperlattice (MSA)
- Heuristics:
  - tree alignments(ClustalW)
  - star alignments
  - sampling (Gibbs) (discussed in RNA2)
  - local profiling with iteration (PSI-Blast, ...)

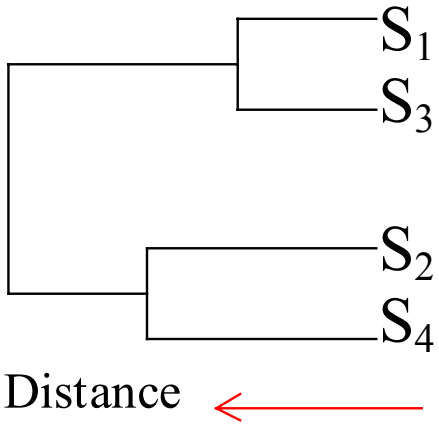
ClustalW: Progressive Multiple Alignment



Similarity Matrix

	S_1	S_2	S_3	S_4
S_1		4	9	4
S_2			4	7
S_3				4
S_4				

Cluster Analysis



Dendrogram

Multiple Alignment Step:
1. Aligning S_1 and S_3
2. Aligning S_2 and S_4
3. Aligning (S_1,S_3) with (S_2,S_4).

See Higgins(1991) and Thompson(1994).

Star Alignments

$s_1 =$ **ATTGCCATT**

$s_2 =$ **ATGGCCATT**

$s_3 =$ **ATCCAATTTT**

$s_4 =$ **ATCTTCTT**

$s_5 =$ **ACTGACC**

Pairwise Alignment

Similarity Matrix

	s_2	s_3	s_4	s_5
s_1	7	−2	0	−3
s_2		−2	0	−4
s_3			0	−7
s_4				−3

Find the Central Sequence s_1

Multiple Alignment

```
ATTGCCATT--
ATGGCCATT--
ATC-CAATTTT
ATCTTC-TT--
ACTGACC----
-----------
AT*GCCATTTT
```

Combine into Multiple Alignment

Pairwise Alignment

```
ATTGCCATT
ATGGCCATT

ATTGCCATT--
ATC-CAATTTT

ATTGCCATT
ATCTTC-TT

ATTGCCATT
ACTGACC
```

DNA2: Today's story and goals

Motivation and connection to DNA1

Comparing types of alignments & algorithms

Dynamic programming

Multi-sequence alignment

Space-time-accuracy tradeoffs

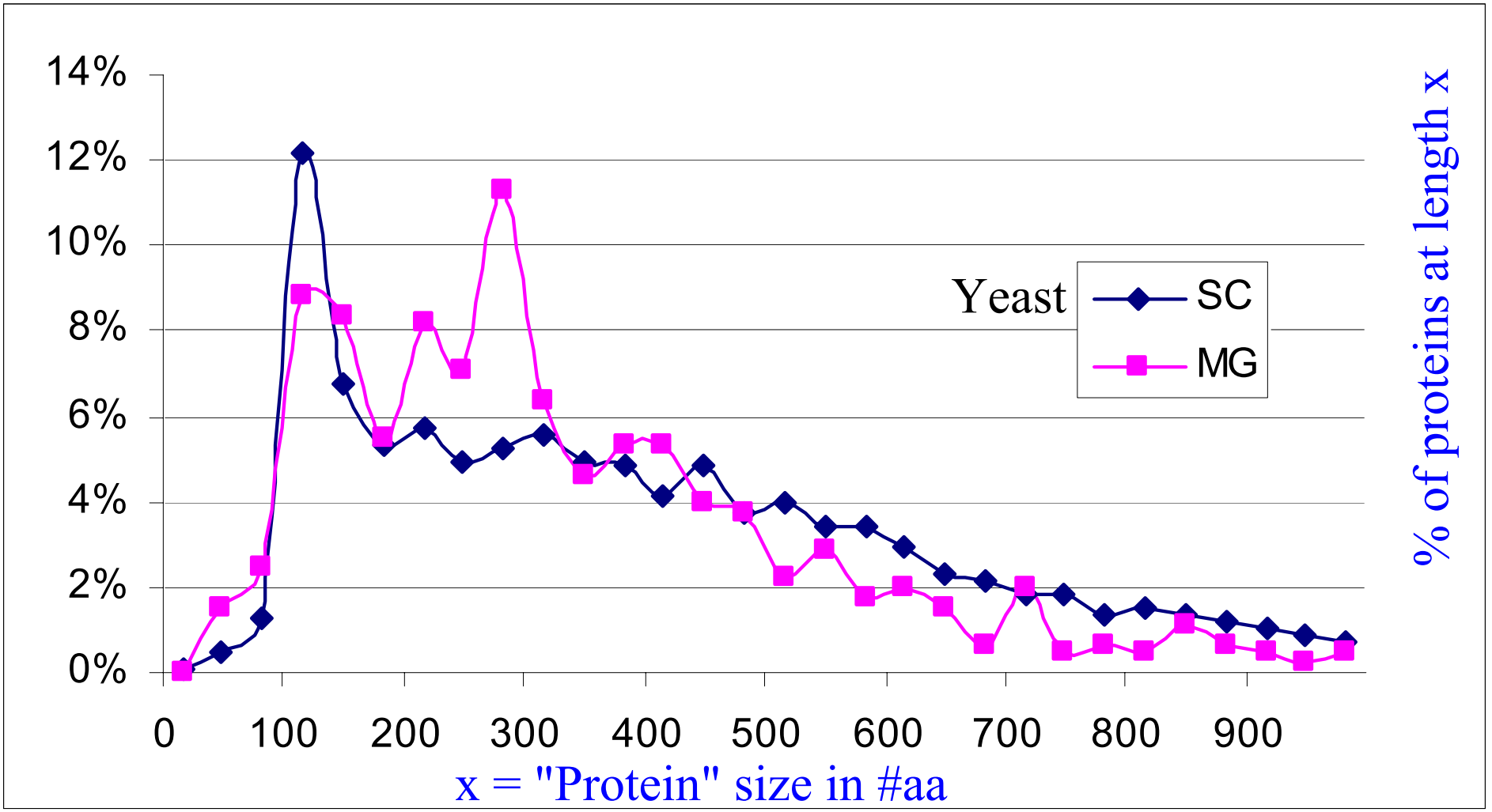
Finding genes -- motif profiles

Hidden Markov Model for CpG Islands

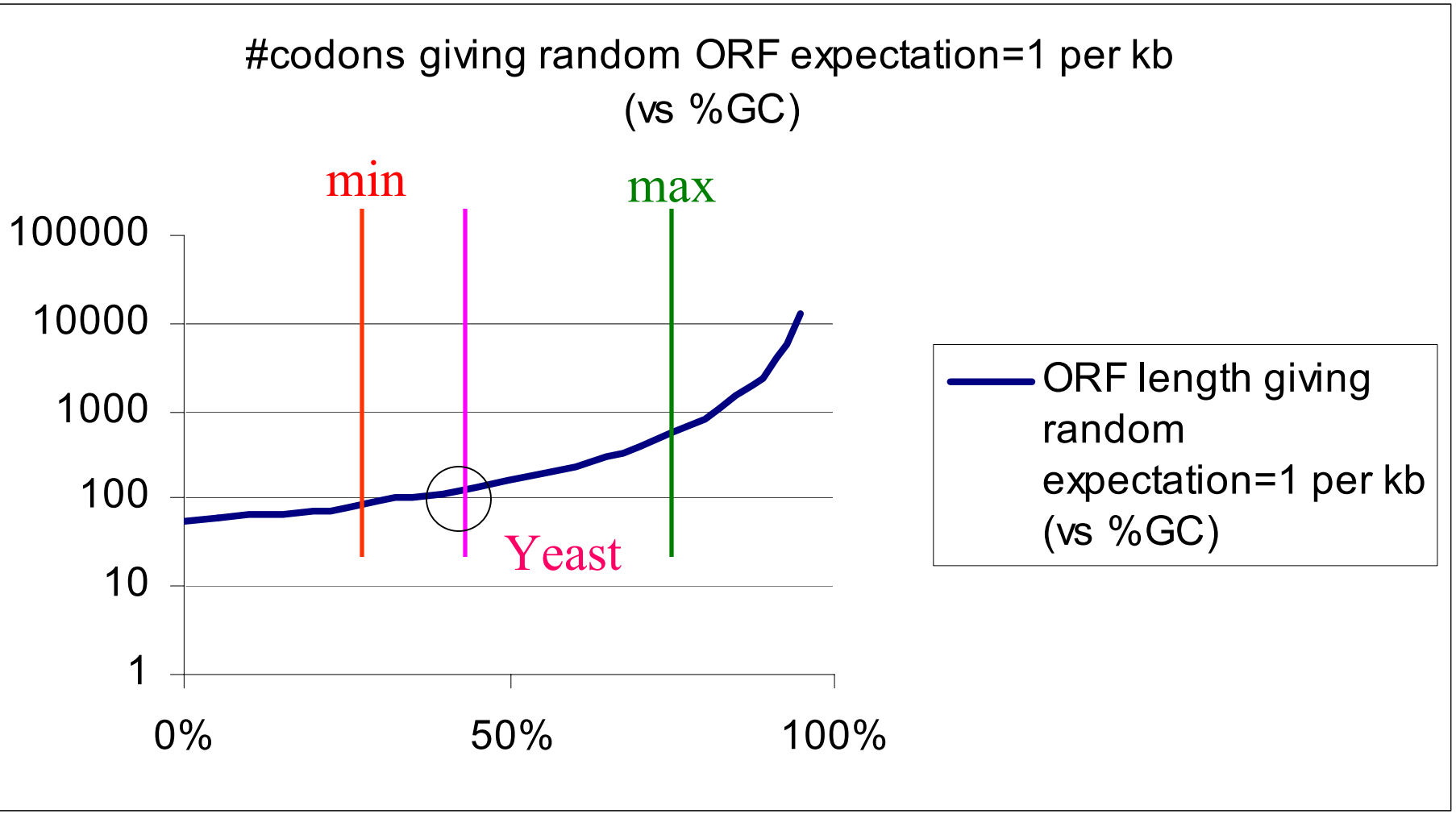
Accurately finding genes & their edges

What is distinctive ?	Failure to find edges?
0. Promoters & CGs islands	Variety & combinations
1. Preferred codons	Tiny proteins (& RNAs)
2. RNA splice signals	Alternatives & weak motifs
3. Frame across splices	Alternatives
4. Inter-species conservation	Gene too close or distant
5. cDNA for splice edges	Rare transcript

Annotated "Protein" Sizes in Yeast & Mycoplasma



Predicting small proteins (ORFs)



Small coding regions

Mutations in domain II of 23 S rRNA facilitate translation of a 23 S rRNA-encoded **pentapeptide** conferring erythromycin resistance. Dam et al. 1996 J Mol Biol 259:1-6

Trp (W) leader peptide, 14 codons:
MKAIFVLKGWWRTS STOP

Phe (F) leader peptide, 15 codons:
MKHIPFFFAFFFTFP STOP

His (H) leader peptide, 16 codons:
MTRVQFKHHHHHHHHPD STOP

Other examples in proteomics lectures

Motif Matrices

a **a t g**
c **a t g**
g **a t g**
t g **t g**

a	1	3	0	0
c	1	0	0	0
g	1	1	0	4
t	1	0	4	0

Align and calculate frequencies.
Note: Higher order correlations lost.

Protein starts

See GeneMark

Motif Matrices

```
   a a t g   1+3+4+4 = 12
   c a t g   1+3+4+4 = 12
   g a t g   1+3+4+4 = 12
   t g t g   1+1+4+4 = 10

a  1 3 0 0
c  1 0 0 0
g  1 1 0 4
t  1 0 4 0

Align and calculate frequencies.
Note: Higher order correlations lost.
Score test sets:
a c c c   1+0+0+0 = 1
```

DNA2: Today's story and goals

Motivation and connection to DNA1

Comparing types of alignments & algorithms

Dynamic programming

Multi-sequence alignment

Space-time-accuracy tradeoffs

Finding genes -- motif profiles

Hidden Markov Model for CpG Islands

Why probabilistic models in sequence analysis?

- **Recognition -** Is this sequence a protein start?
- **Discrimination -** Is this protein more like a hemoglobin or a myoglobin?
- **Database search -** What are all of sequences in SwissProt that look like a serine protease?

A Basic idea

Assign a number to every possible sequence such that

$$\sum_s P(s|M) = 1$$

P(s|M) is a probability of sequence s given a model M.

Sequence recognition

Recognition question **- What is the probability that the sequence s is from the start site model M ?**

P(M|s) = P(M)* P(s|M) / P(s)

(Bayes' theorem)

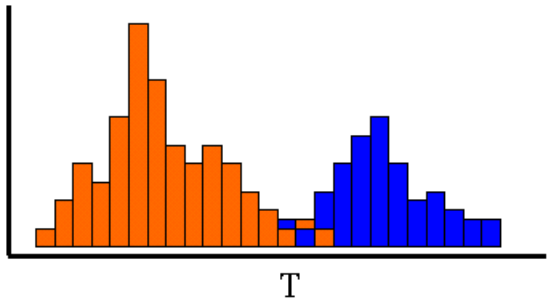
P(M) and P(s) are prior probabilities and P(M|s) is posterior probability.

Database search

- **N = null model (random bases or AAs)**
- **Report all sequences with**

$$\log P(s|M) - \log P(s|N) > \log P(N) - \log P(M)$$

- **Example, say α/β hydrolase fold is rare in the database, about 10 in 10,000,000. The threshold is 20 bits. If considering 0.05 as a significant level, then the threshold is 20+4.4 = 24.4 bits.**



Plausible sources of mono, di, tri, & tetra- nucleotide biases

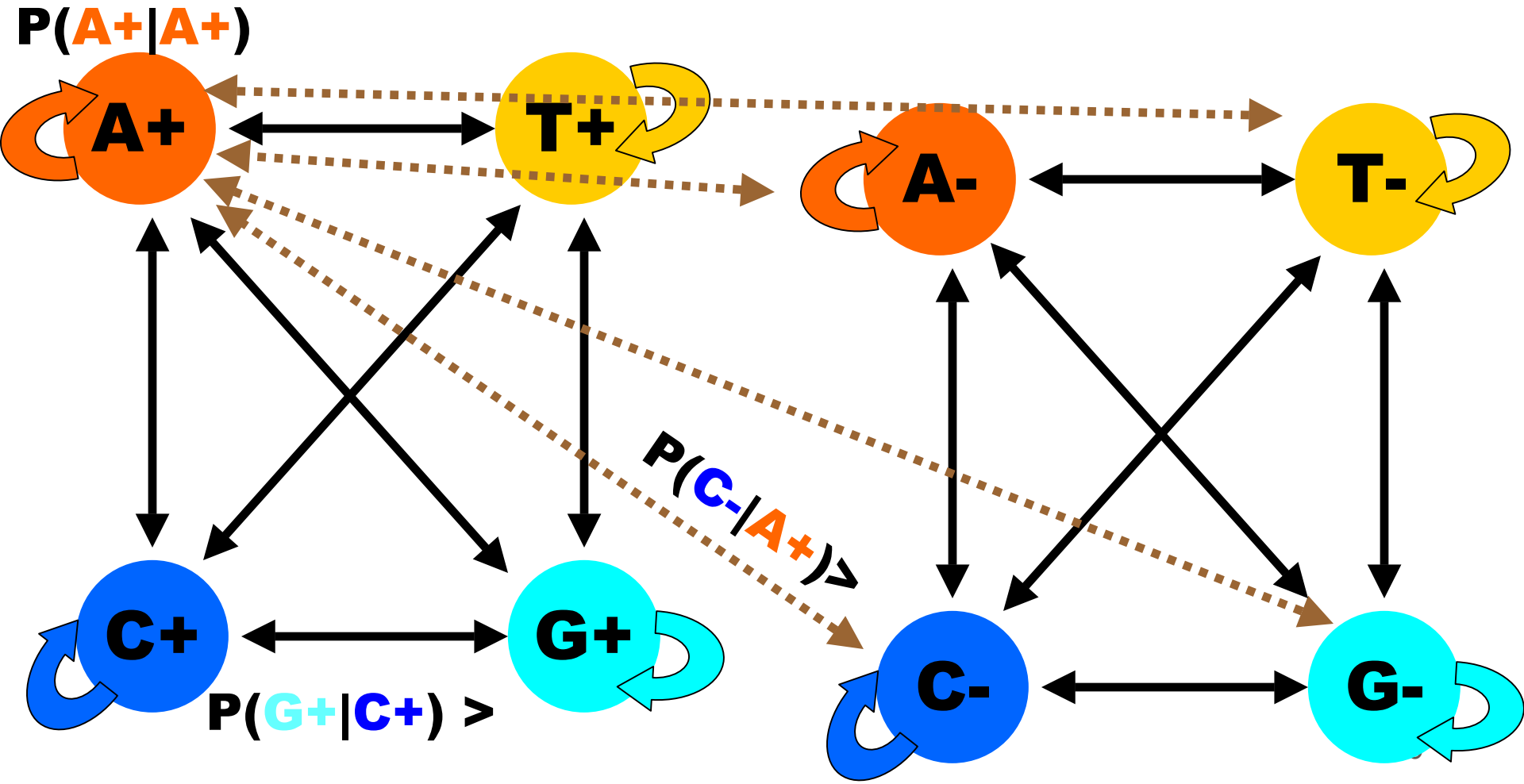
C rare due to lack of uracil glycosylase (cytidine deamination)
TT rare due to lack of UV repair enzymes.
CG rare due to 5methylCG to TG transitions (cytidine deamination)
AGG rare due to low abundance of the corresponding Arg-tRNA.
CTAG rare in bacteria due to error-prone "repair" of CTAGG to C*CAGG.
AAAA excess due to polyA pseudogenes and/or polymerase slippage.

AmAcid	Codon	Number	/1000	Fraction
Arg	AGG	3363.00	1.93	0.03
Arg	AGA	5345.00	3.07	0.06
Arg	CGG	10558.00	6.06	0.11
Arg	CGA	6853.00	3.94	0.07
Arg	CGT	34601.00	19.87	0.36
Arg	CGC	36362.00	20.88	0.37

ftp://sanger.otago.ac.nz/pub/Transterm/Data/codons/bct/Esccol.cod

CpG Island + in a ocean of -
First order Hidden Markov Model

MM=16, HMM= 64 transition probabilities (adjacent bp)



Estimate transistion probabilities -- an example

Training set

S= aaacagcctgacatggttc**CGAACAGCCTCGACATGGCGTT**

Isle	A+	C+	G+	T+	A-	C-	G-	T-	
A+	.20	.40	.20	.20	.00	.00	.00	.00	1.00
C+	.29	.14	.43	.14	.00	.00	.00	.00	1.00
G+	.33	.33	.17	.17	.00	.00	.00	.00	1.00
T+	.00	.25	.25	.25	.25	.00	.00	.00	1.00
	.82	1.13	1.05	.76	.25	.00	.00	.00	**Sums**
Ocean	**A-**	**C-**	**G-**	**T-**	**A+**	**C+**	**G+**	**T+**	
A-	.33	.33	.17	.17	.00	.00	.00	.00	1.00
C-	.40	.20	.00	.20	.00	.20	.00	.00	1.00
G-	.25	.25	.25	.25	.00	.00	.00	.00	1.00
T-	.00	.25	.50	.25	.00	.00	.00	.00	1.00
	.98	1.03	.92	.87	.00	.20	.00	.00	**Sums**

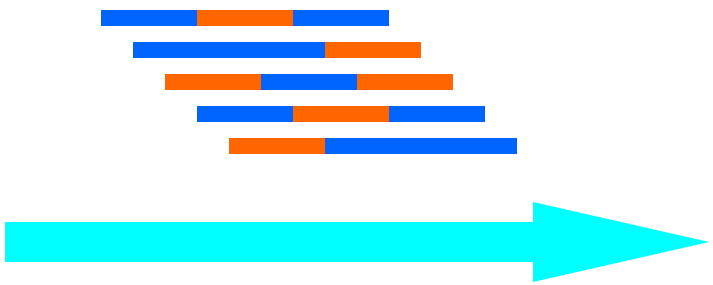
$P(G|C) = \#(CG) / \sum_N \#(CN)$

Laplace pseudocount: Add +1 count to each observed. (p.9,108,321 Dirichlet)

(http://shop.barnesandnoble.com/textbooks/booksearch/isbninquiry.asp?mscssid=&isbn=0521629713)

Estimated transistion probabilities from 48 "known" islands

Training set



$$P(G|C) = \#(CG) / \sum_N \#(CN)$$

(+)	A	C	G	T	
A	.18	.27	.43	.12	1.00
C	.17	.37	.27	.19	1.00
G	.16	.34	.38	.13	1.00
T	.08	.36	.38	.18	1.00

(-)	A	C	G	T	
A	.30	.21	.29	.21	1.00
C	.32	.30	.08	.30	1.00
G	.25	.25	.30	.21	1.00
T	.18	.24	.29	.29	1.00
	1.05	.99	.95	1.01	**Sums**

Viterbi: dynamic programming for HMM

1/8*.27

Most probable path

l,k=2 states

v	S_i =	C	G	C	G
begin	1	0	0	0	0
A+	0	0	0	0	0
C+	0	**0.125**	0	**0.012**	0
G+	0	0	**0.034**	0	**0.0032**
T+	0	0	0	0	0
A-	0	0	0	0	0
C-	0	0.125	0	0.0026	0
G-	0	0	0.01	0	0.0002
T-	0	0	0	0	0

Recursion:

$v_l(i+1) =$

$e_l(x_i+1)\ \max(v_k(i)a_{kl})$

a= table in slide 51

e= emit S_i in state l (Durbin p.56)

DNA2: Today's story and goals

Motivation and connection to DNA1

Comparing types of alignments & algorithms

Dynamic programming

Multi-sequence alignment

Space-time-accuracy tradeoffs

Finding genes -- motif profiles

Hidden Markov Model for CpG Islands