Genome annotation

- Protein-coding genes
- Non-coding genes (effector and regulatory RNAs)
- Regulatory elements

AGTAGTGTGTGCCCGTCTGTTGTGTGACTCTGGTAGCTAGAGATCCCTCAGACCCTTTGTGGTAGTGTGGA
AAATCTCTAGCAGTGGCGCCCGAACAGGGACTTAAAAGCGAAAGTAAGACCAGAGGAGATCTCTCGACGCA
GGACTCGGCTTGCTGAAGTGCACTCGGCAAGAGGCGAGAGGGGCGGCTGGTGAGTACGCCATTTTTTATTT
GACTAGCGGAGGCTAGAAGGAGAGAGATGGGTGCGAGAGCGTCAATATTAAGAGGCGAAAAATTAGATAAA
TGGGAAAGAATTAGGTTAAGGCCAGGGGGAAAGAAAAGCTATATGATATAGCACTTAATATGGGCAAGCAG
GGAGCTGGAAAGATTTGCACTCAACTCTGGCCTTTTAGAAACATCAGGAGGCTGTAAACAAATAATGAAAC
AGCTACAACCAGCTCTACAGACAGGAACAGAGGAACTTAAATCATTATATAACACAGTAGCAACTCTCTAT
TGTGTACATGAAAAAATAGAAGTACGAGACACCAAGGAAGCCTTAGACAAGATAGAGGAAGAACAAAACAA
AAGTCAGCAAAAAACACAGCAGGCAGCTGACGGAAAGGTCAGTCAAAATTATCCTATAGTGCAGAATCTTC
AAGGGCAAATGGTACATCAAGCCATATCACCTAGAACCTTGAATGCATGGGTAAAAGTAATAGAGGAGAAG
GCTTTTAGCCCAGAGGTAATACCCATGTTTACAGCATTATCAGAAGGAGCCACCCCACAAGATTTAAACAC
CATGTTAAATACGGTGGGGGGACATCAAGCAGCCATGCAAATGTTAAAGGATACCATCAATGAAGAGGCTG
CAGAATGGGATAGATTACATCCAGTACATGCGGGGCCTATTGCACCAGGCCAAATGAGAGAACCAAGGGGA
AGTGACATAGCAGGAACTACTAGTACCCTTCAGGAACAAATATCATGGATAACAGGTAACCCACCTATTCC
AGTGGGAGAAATCTATAAAAGATGGATAATTCTGGGGTTAAACAAAATAGTGAGAATGTATAGCCCTGTCA
GCATTTTGGACATAAGACAAGGGCCAAAGGAACCCTTTAGAGACTATGTAGATCGGTTCTTTAAAACTTTA
AGAGCTGAACAAGCTACACAAGATGTAAAAAATTGGATGACAGACACCTTGTTGGTCCAAAATGCGAACCC
AGATTGTAAGACCATTTTAAGAGCATTAGGACCAGGGGCCACATTAGAAGAAATGATGACAGCATGTCAGG
GAGTGGGAGGACCTGGCCACAAAGCAAGAGTGTTGGCTGAGGCAATGAGCCAAGCTAACAATATAAACATA
ATGATGCAGAGAAGCAATTTTAAAGGCTCTAAGAGAACTATTAAATGTTTCAACTGCGGCAAGGAAGGGCA
CATAGCTAGAAACTGCAGGGCCCCTAGGAAAAAAGGCTGTTGGAAATGTGGTAAGGAAGGACACCAAATGA
AAGACTGTACTGAGAGGCAGGCTAATTTTTTAGGGAAAATTTGGCCTTCCCAGAAGGGGAGGCCAGGGAAT
TTCCTTCAGAGCAGACCAGAGCCAACAGCCCCACCAGCAGAGAGCTTCAAGTTCGAGGAGACAACCCCCGT
TCCGAAGCAGGAGCCGAAAGACAGGGAACCCTTAACTTCCCTCAAAT…

What is “written” in here?

AGTAGTGTGTGCCCGTCTGTTGTGTGACTCTGGTAGCTAGAGATCCCTCAGACCCTTTGTGGTAGTGTGGA
AAATCTCTAGCAGTGGCGCCCGAACAGGGACTTAAAAGCGAAAGTAAGACCAGAGGAGATCTCTCGACGCA
GGACTCGGCTTGCTGAAGTGCACTCGGCAAGAGGCGAGAGGGGCGGCTGGTGAGTACGCCATTTTTTATTT
GACTAGCGGAGGCTAGAAGGAGAGAGATGGGTGCGAGAGCGTCAATATTAAGAGGCGAAAAATTAGATAAA
TGGGAAAGAATTAGGTTAAGGCCAGGGGGAAAGAAAAGCTATATGATATAGCACTTAATATGGGCAAGCAG
GGAGCTGGAAAGATTTGCACTCAACTCTGGCCTTTTAGAAACATCAGGAGGCTGTAAACAAATAATGAAAC
AGCTACAACCAGCTCTACAGACAGGAACAGAGGAACTTAAATCATTATATAACACAGTAGCAACTCTCTAT
TGTGTACATGAAAAAATAGAAGTACGAGACACCAAGGAAGCCTTAGACAAGATAGAGGAAGAACAAAACAA
AAGTCAGCAAAAAACACAGCAGGCAGCTGACGGAAAGCTCAGTCAAAATTATCCTATAGTGCAGAATCTTC
AAGGGCAAATGGTACATCAAGCCATATCACCTAGAACTTGAATGCATGGGTAAAAGTAATAGAGGAGAAG
GCTTTTAGCCCAGAGGTAATACCCATGTTTAGCATTATCAGAAGGAGCCACCCCACAAGATTTAAACAC
CATGTTAAATACGGTGGGGGGACATCAAGCAGCCATGCAAATGTTAAAGGATACCATCAATGAAGAGGCTG
CAGAATGGGATAGATTACATCCAGTACATGCGGGGCCTATTGCACCAGGCCAAATGAGAGAACCAAGGGGA
AGTGACATAGCAGGAACTACTAGTACCCTTCAGGAACAAATATCATGGATAACAGGTAACCCACCTATTCC
AGTGGGAGAAATCTATAAAAGATGGATAATTCTGGGGTTAAACAAAATAGTGAGAATGTATAGCCCTGTCA
GCATTTTGGACATAAGACAAGGGCCAAAGGAACCCTTTAGAGACTATGTAGATCGGTTCTTTAAAACTTTA
AGAGCTGAACAAGCTACACAAGATGTAAAAAATTGGATGACAGACACCTTGTTGGTCCAAAATGCGAACCC
AGATTGTAAGACCATTTTAAGAGCATTAGGACCAGGGGCCACATTAGAAGAAATGATGACAGCATGTCAGG
GAGTGGGAGGACCTGGCCACAAAGCAAGAGTGTTGGCTGAGGCAATGAGCCAAGCTAACAATATAAACATA
ATGATGCAGAGAAGCAATTTTAAAGGCTCTAAGAGAACTATTAAATGTTTCAACTGCGGCAAGGAAGGGCA
CATAGCTAGAAACTGCAGGGCCCCTAGGAAAAAAGGCTGTTGGAAATGTGGTAAGGAAGGACACCAAATGA
AAGACTGTACTGAGAGGCAGGCTAATTTTTTAGGGAAAATTTGGCCTTCCCAGAAGGGGAGGCCAGGGAAT
TTCCTTCAGAGCAGACCAGAGCCAACAGCCCCACCAGCAGAGAGCTTCAAGTTCGAGGAGACAACCCCCGT
TCCGAAGCAGGAGCCGAAAGACAGGGAACCCTTAACTTCCCTCAAAT…

GAG gene

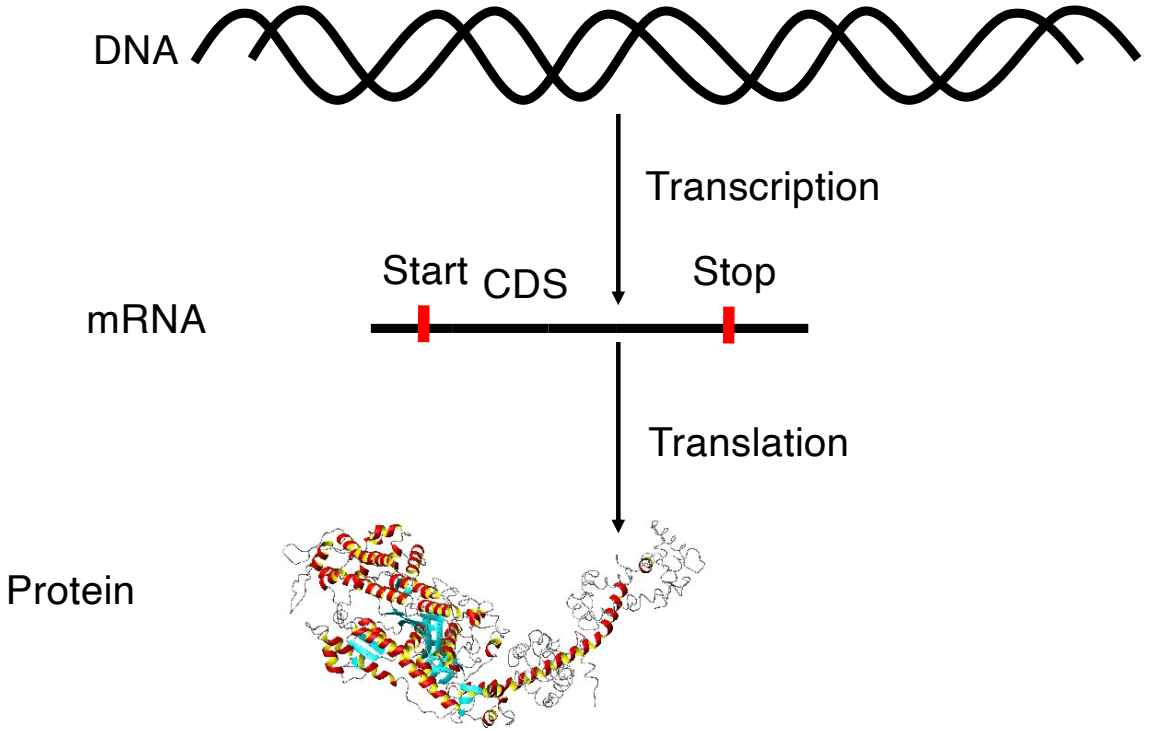
What is “written” in here?

Approach: describe what a gene is with a Markov model and use the model to identify genes in genomic sequences.

What is emitted is the DNA sequence but what are the “hidden” states?

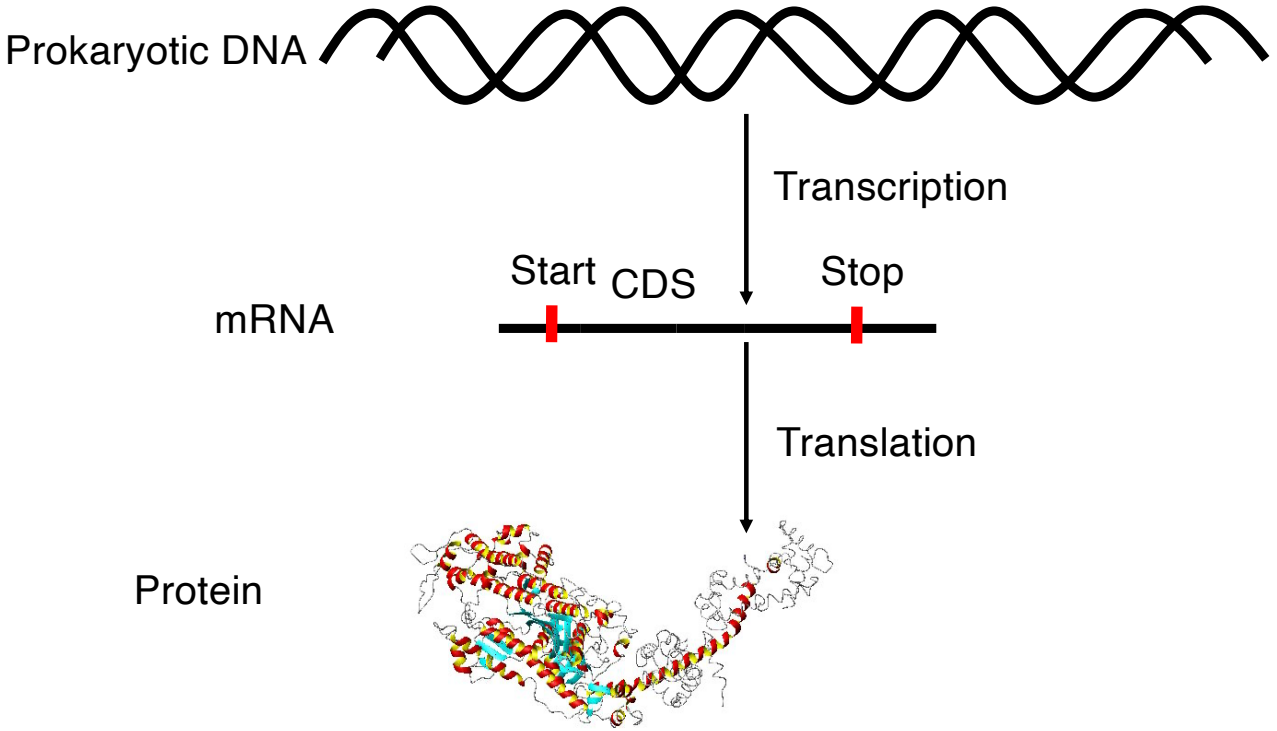
What is a gene?

Contiguous stretch of nucleotides in the genome that encodes a protein.



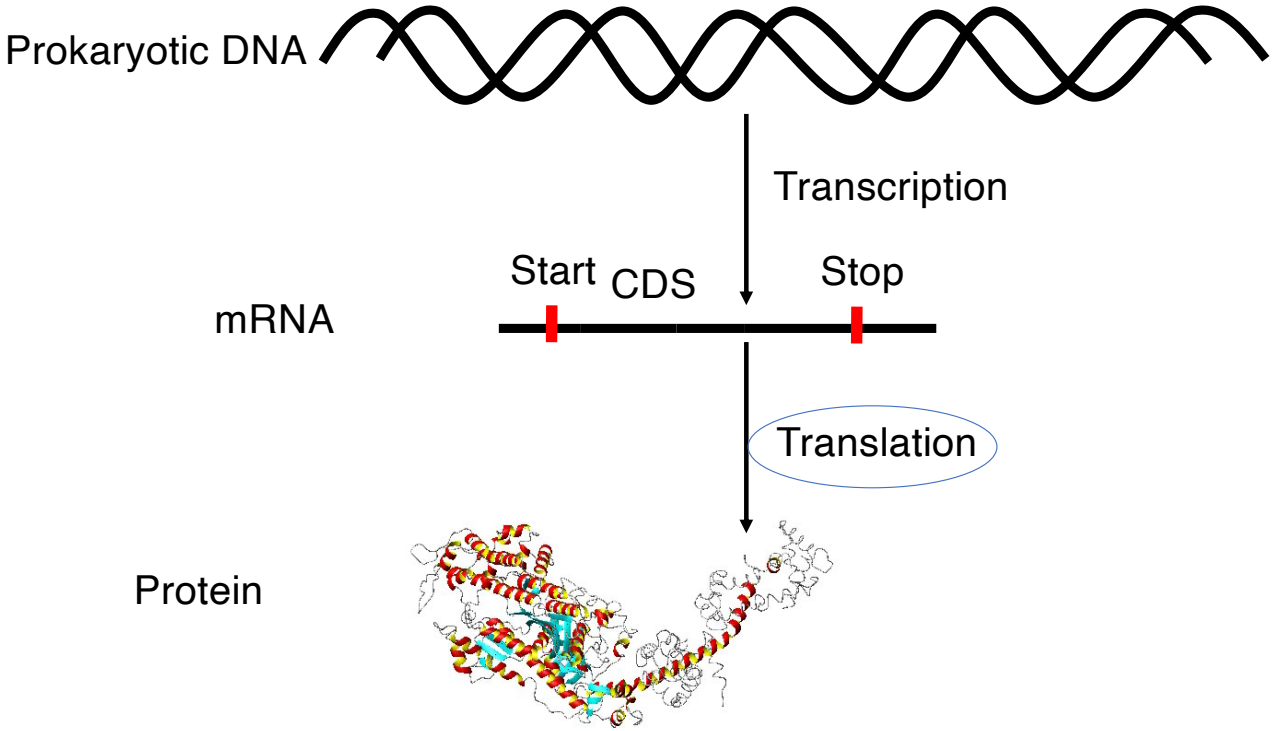
What is a prokaryotic gene?

Contiguous stretch of nucleotides in the genome that encodes a protein.

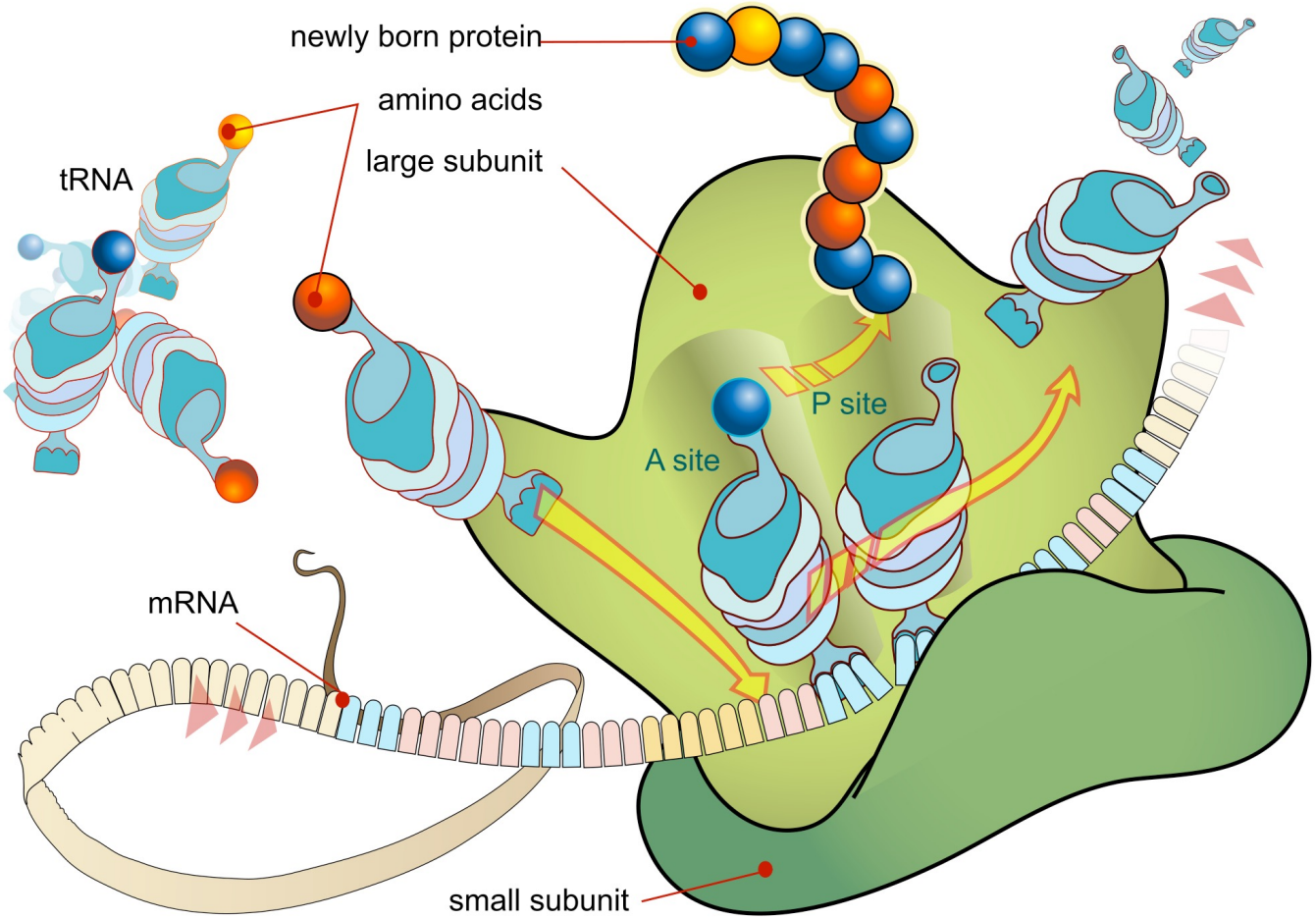


What is a prokaryotic gene?

Contiguous stretch of nucleotides in the genome that encodes a protein.

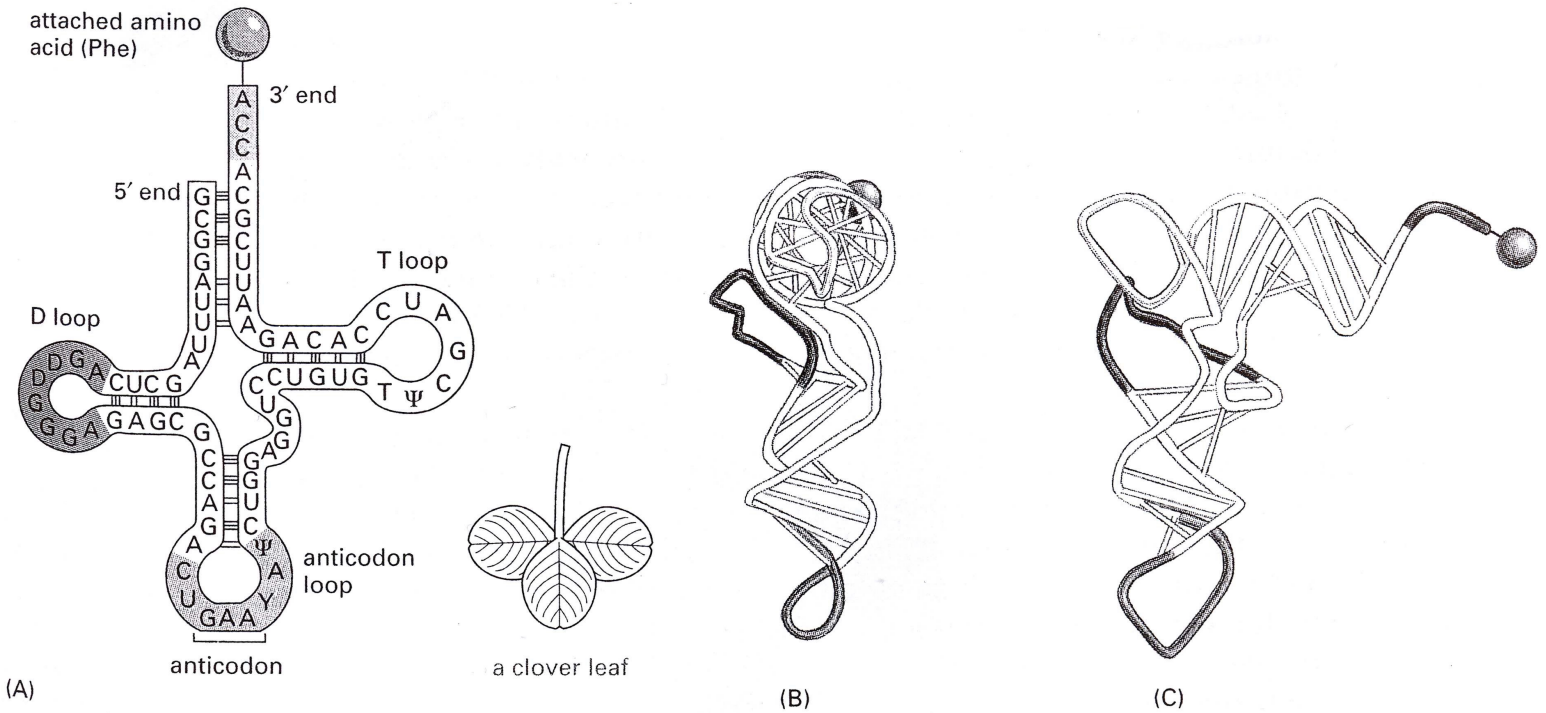


mRNA translation



www.wikipedia.org

tRNAs in translation

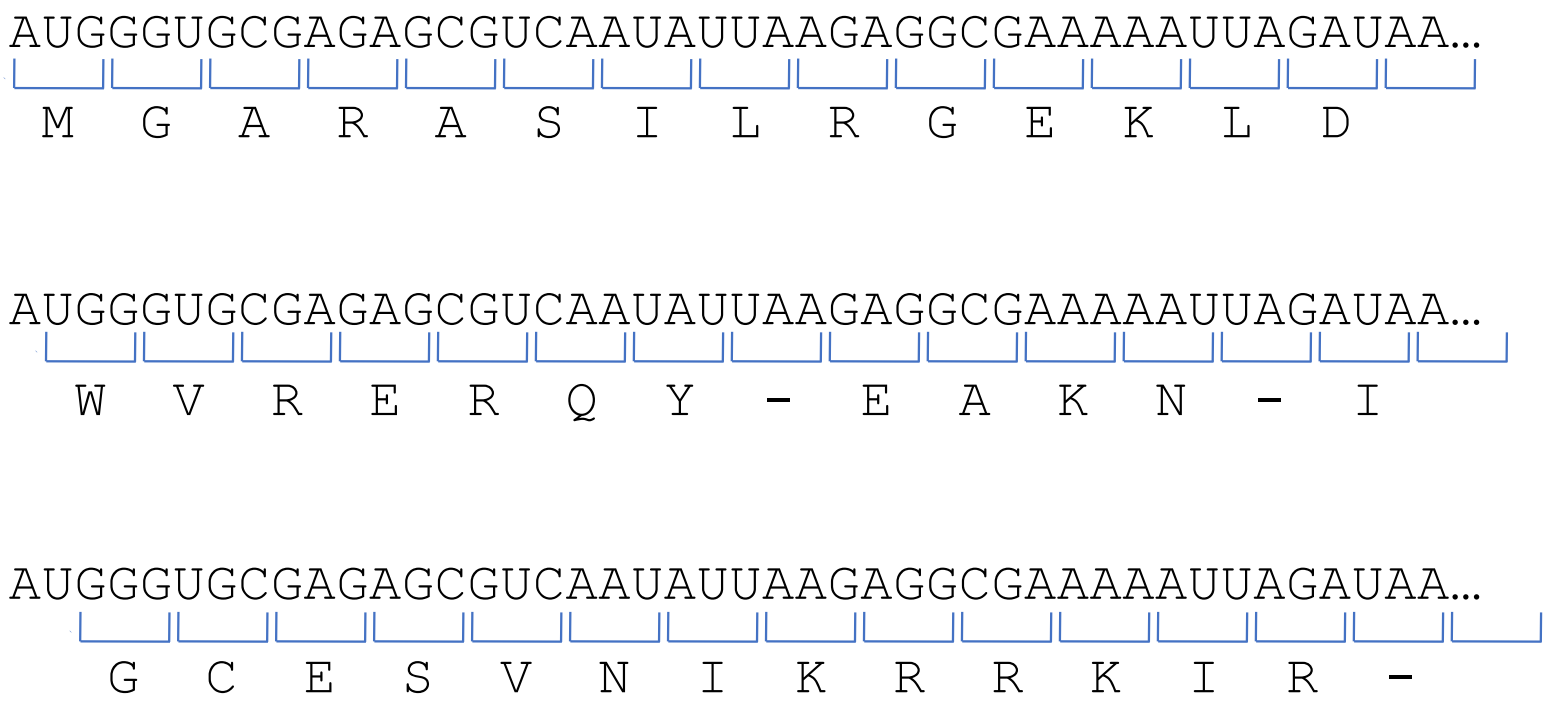


Codon Table

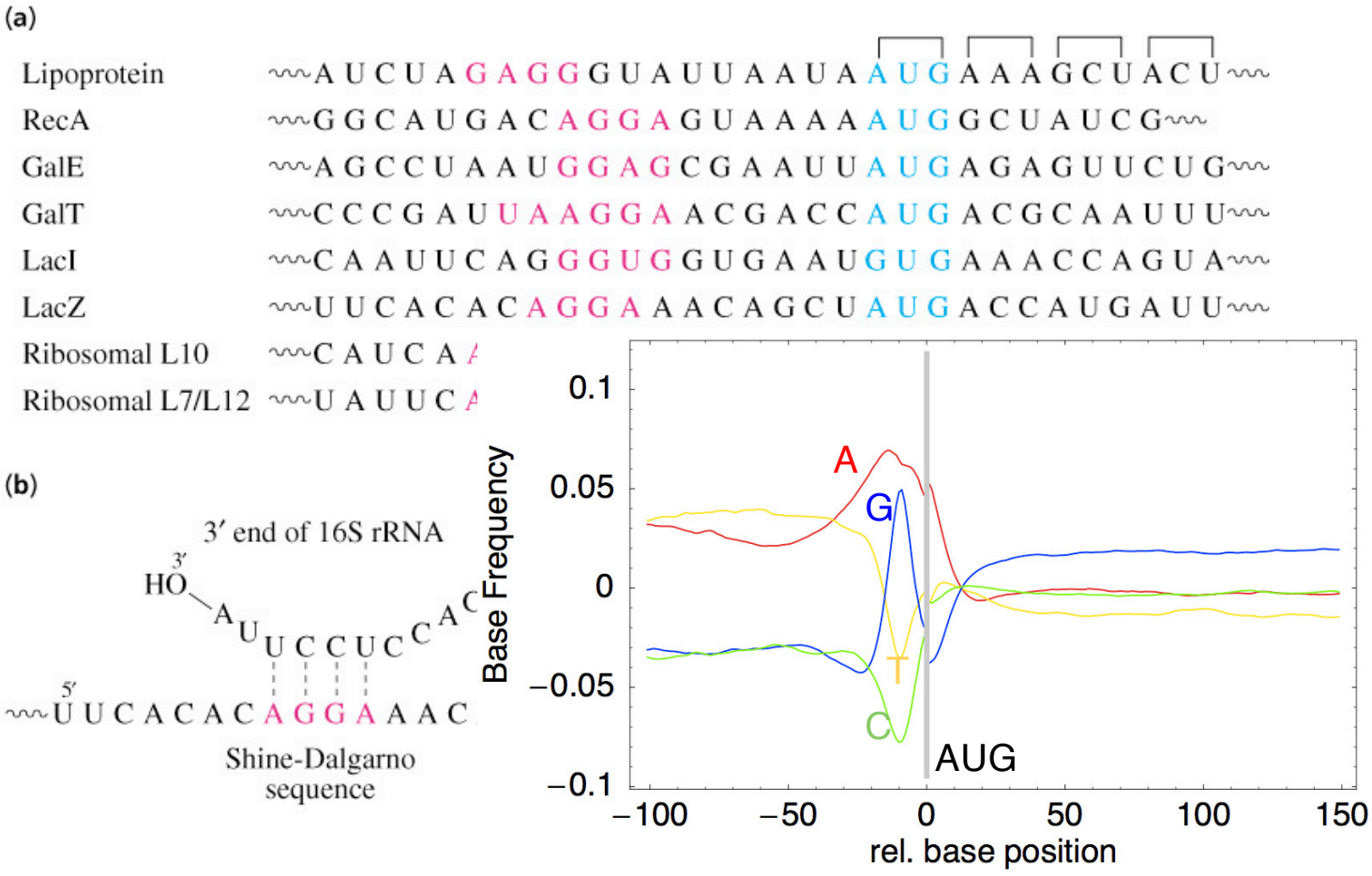
Second nucleotide

First nucleotide	U	C	A	G	Third nucleotide
U	UUU Phe UUC Phe UUA Leu UUG Leu	UCU Ser UCC Ser UCA Ser UCG Ser	UAU Tyr UAC Tyr UAA Stop UAG Stop	UGU Cys UGC Cys UGA Stop UGG Trp	U C A G
C	CUU Leu CUC Leu CUA Leu CUG Leu	CCU Pro CCC Pro CCA Pro CCG Pro	CAU His CAC His CAA Gln CAG Gln	CGU Arg CGC Arg CGA Arg CGG Arg	U C A G
A	AUU Ile AUC Ile AUA Ile AUG Met	ACU Thr ACC Thr ACA Thr ACG Thr	AAU Asn AAC Asn AAA Lys AAG Lys	AGU Ser AGC Ser AGA Arg AGG Arg	U C A G
G	GUU Val GUC Val GUA Val GUG Val	GCU Ala GCC Ala GCA Ala GCG Ala	GAU Asp GAC Asp GAA Glu GAG Glu	GGU Gly GGC Gly GGA Gly GGG Gly	U C A G

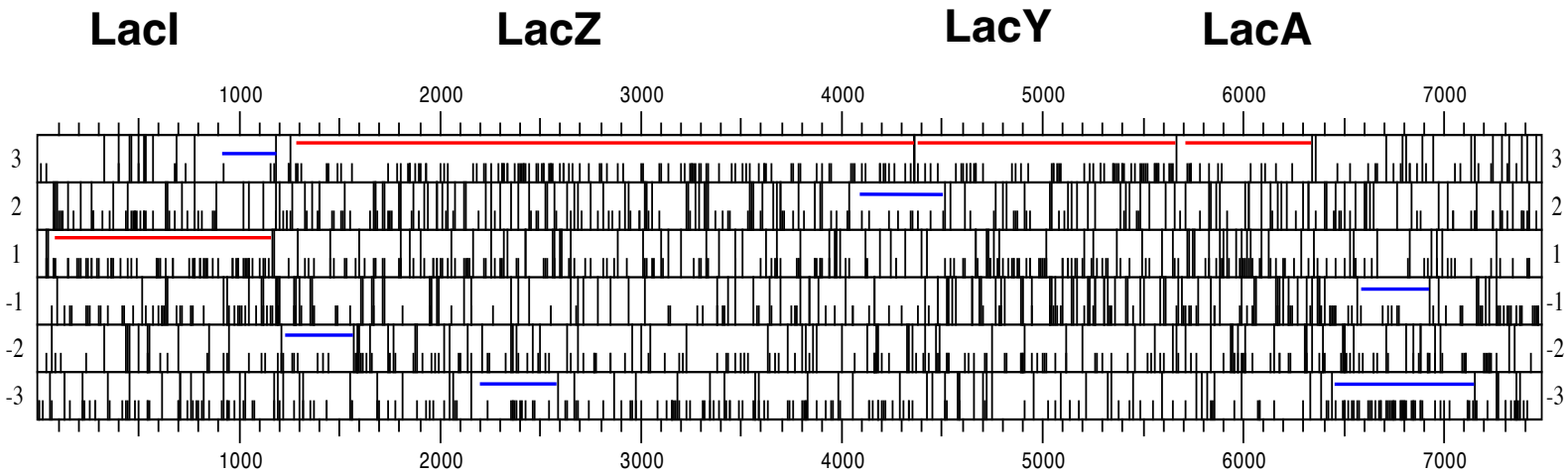
Open reading frame



How does the ribosome know where the start is?

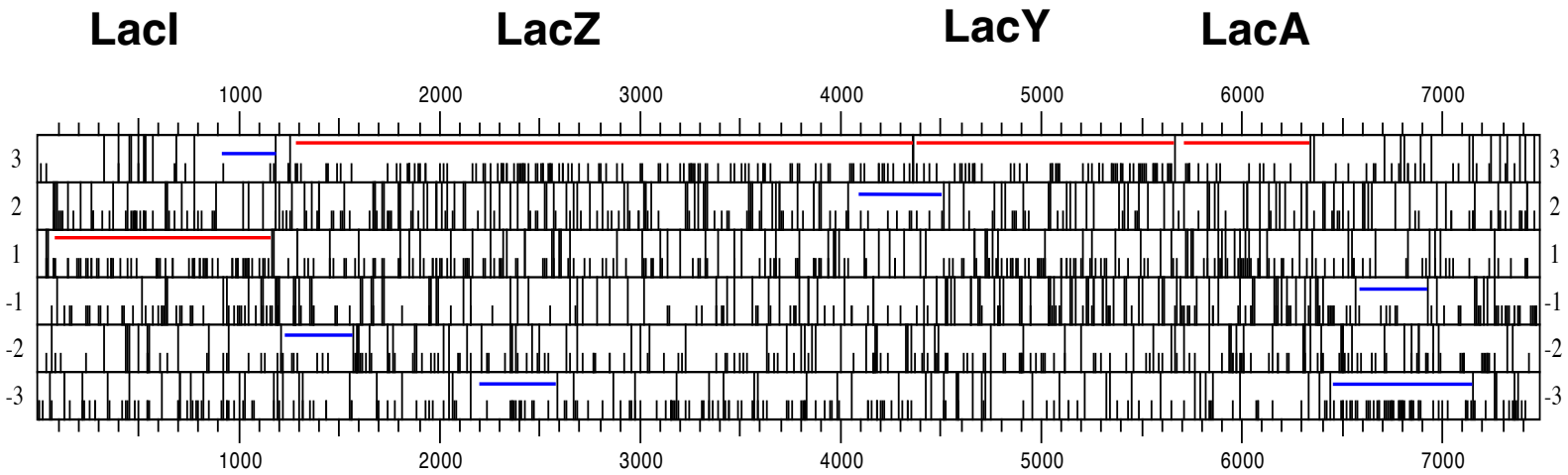


Is the recognition of open reading frames sufficient?



Location of start (short vertical lines) and stop (long vertical lines) within the E.coli LacZ operon

Is the recognition of open reading frames sufficient?



How do we distinguish real from “spurious” ORFs?

Length?!

ORFs expected “by chance”

• Imagine that we generate random ORFs by choosing at each step one of the 64 possible codons with equal probability.

• What is the average length of an ORFs generated by this process?

• Calculation is similar to the one we used to determine the expected length of CpG islands:

$$P(Stop) = P(UAA \text{ or } UAG \text{ or } UGA) = \frac{3}{64}$$

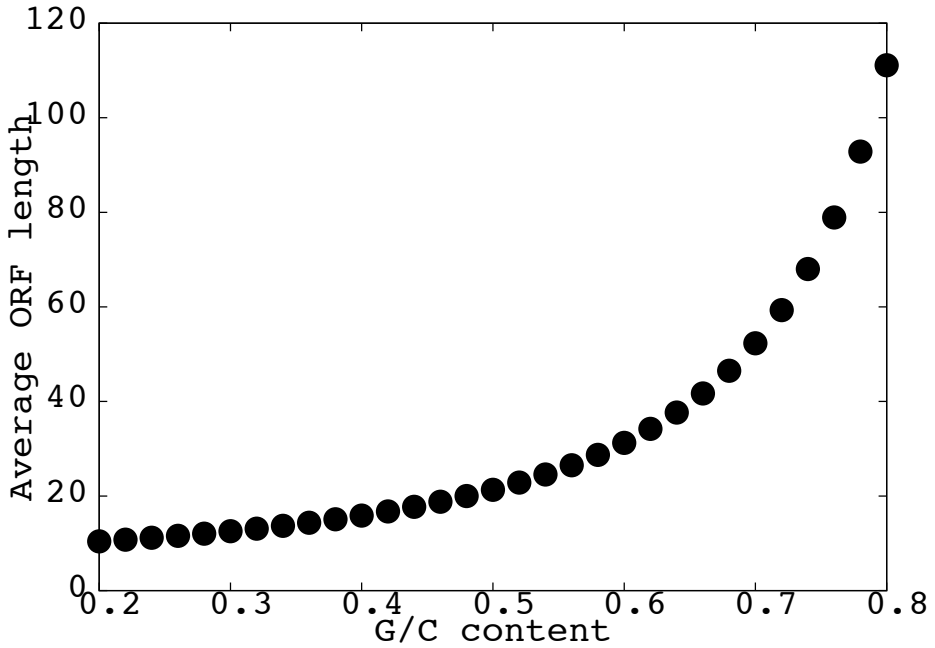
$$P((\overline{Stop})_{l-1}\, Stop) = \left(1 - \frac{3}{64}\right)^{l-1} \frac{3}{64} \quad \text{Let } p = 1 - \frac{3}{64}$$

$$\text{Then } < l >= \sum_{l=1}^{\infty} l p^{l-1}(1-p) = (1-p)\frac{\partial}{\partial p}\sum_{l=1}^{\infty} p^{l}$$

$$< l >= (1-p)\frac{1}{(1-p)^2} = \frac{1}{1-p} = \frac{64}{3}$$

ORFs expected “by chance”

Assuming that the proportions of G’s and C’s in a genome are identical and that so are the proportions of A’s and T’s, what is the average ORF length that we expect in genomes of varying G/C contents?



Why look beyond “long ORFs”?

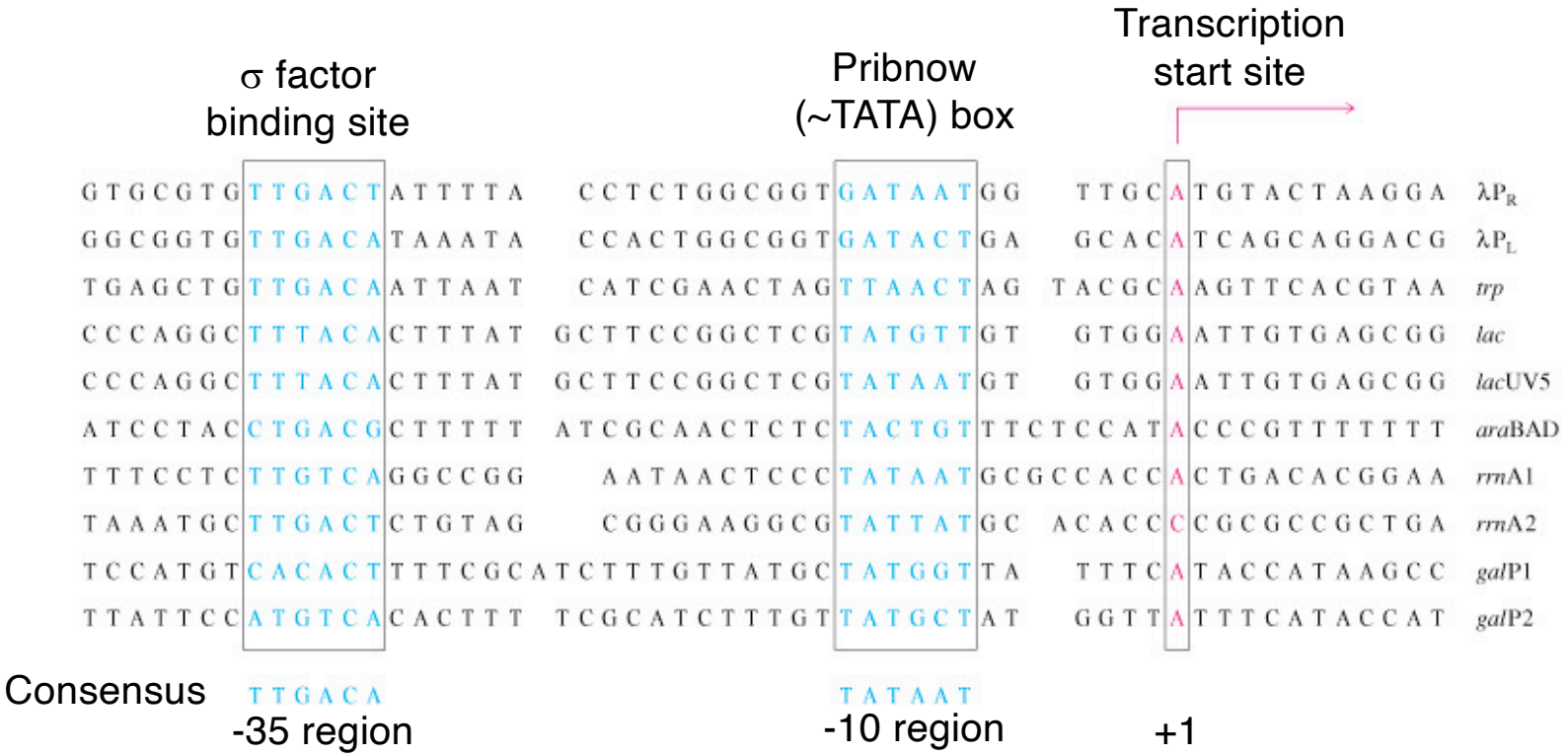
- We would still need to choose a cutoff for deciding that the ORFs are long enough for their occurrence to not be due to “chance”. Cutoff depends on factors like the G/C content of the genome.
- There really are short ORFs that encode functional peptides which we would miss if we set hard cutoff on ORF length.
- There are also genes that do not encode proteins.

Moreover…

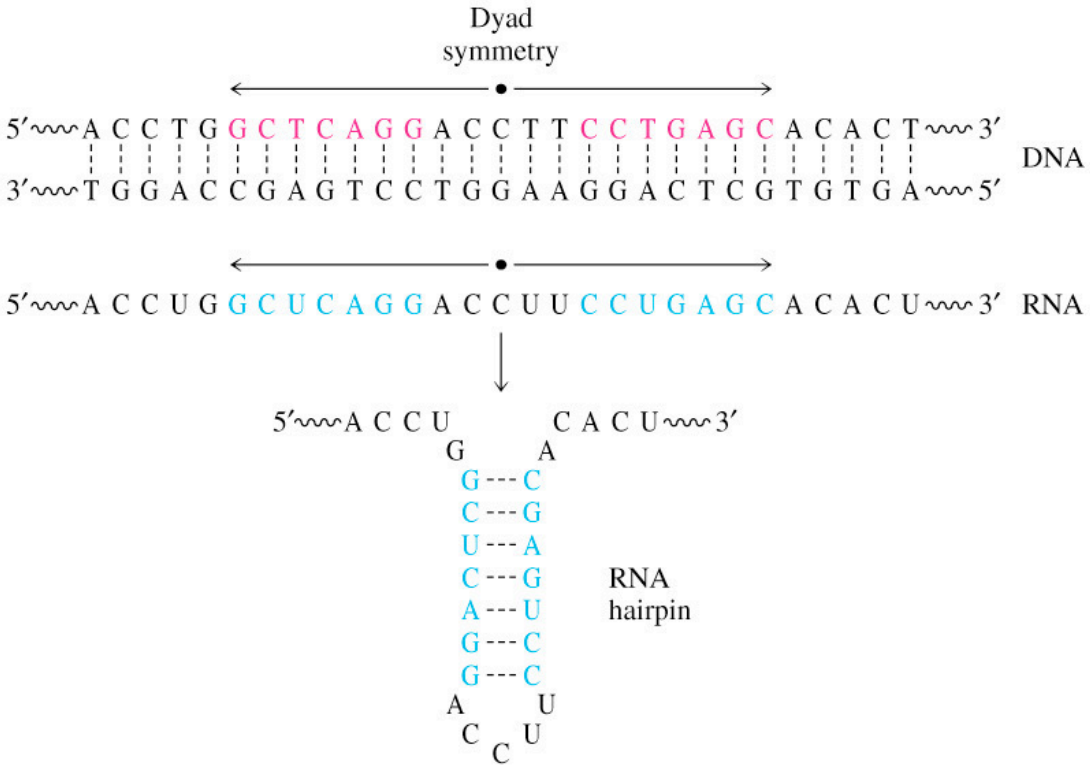
We do have more information about what makes a protein-coding gene and this information may help us identify even short ORFs.

Incorporating additional signals

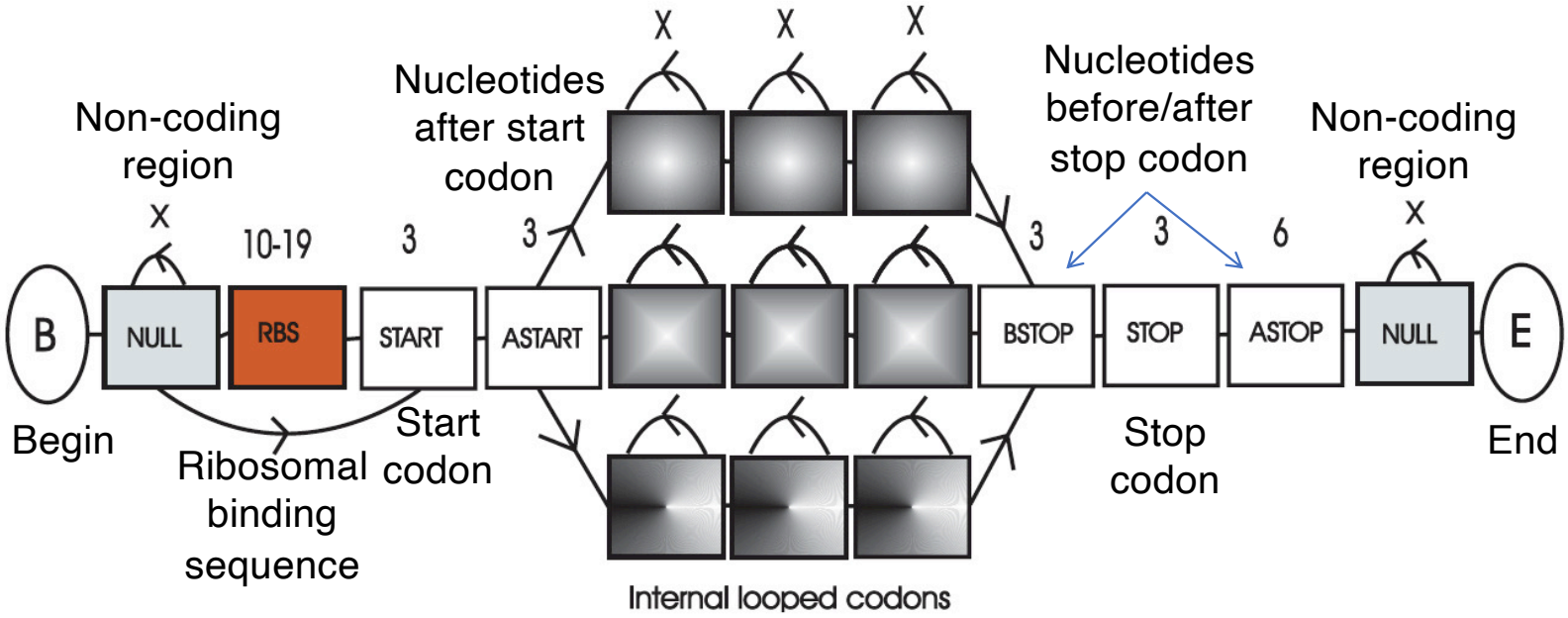
Promoter regions recognized by the σ70 subunit of E.coli



Transcription terminators

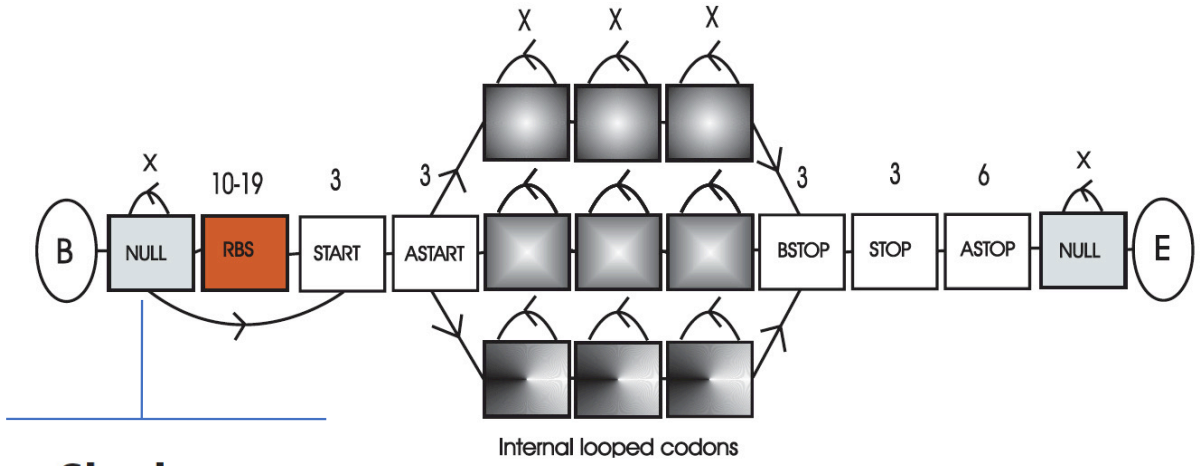


A realistic HMM for bacterial gene prediction

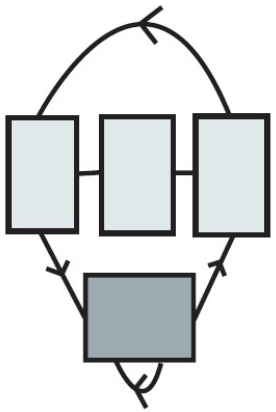


EasyGene
Larsen & Krogh. BMC Bioinformatics 2003

Null model



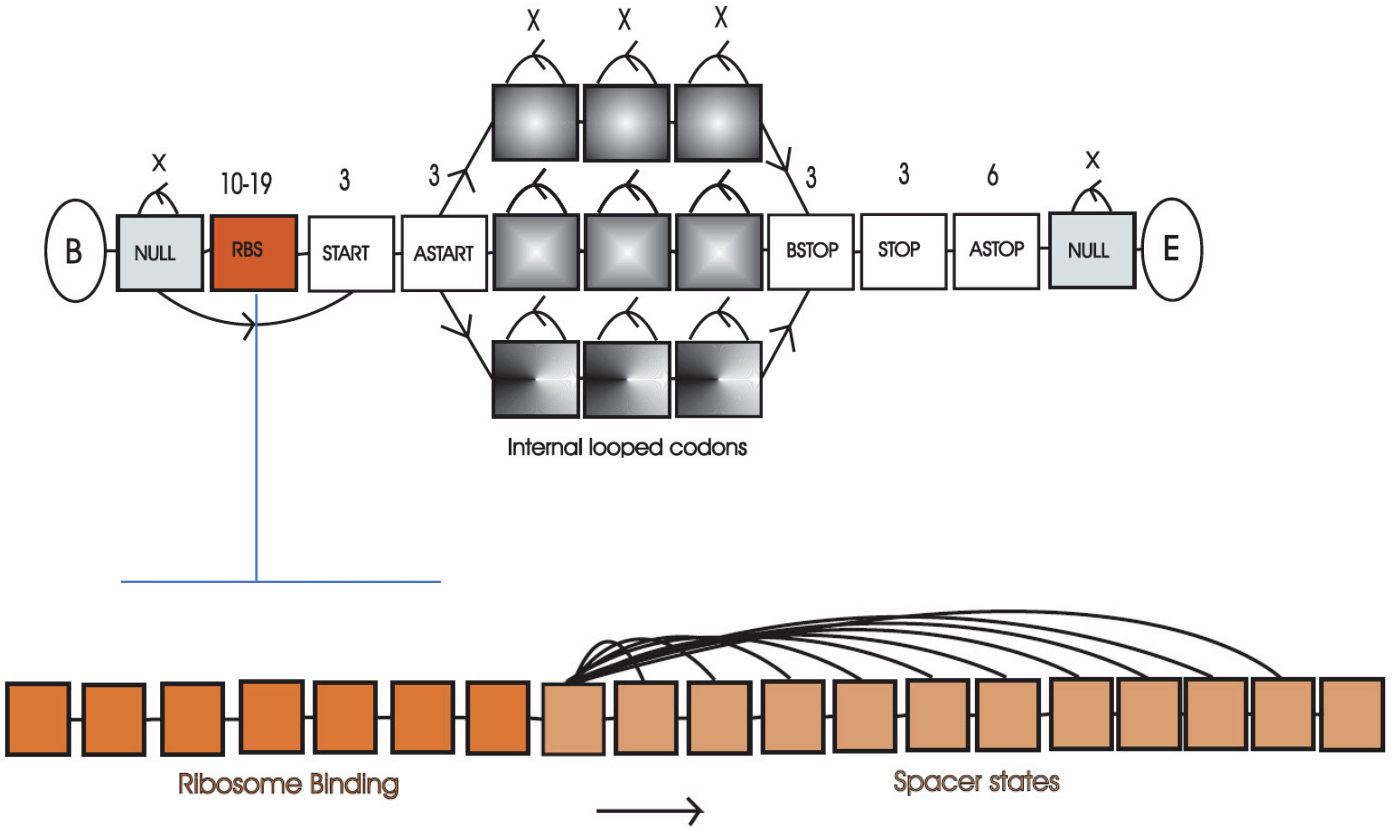
Shadows



Coding region on the opposite strand

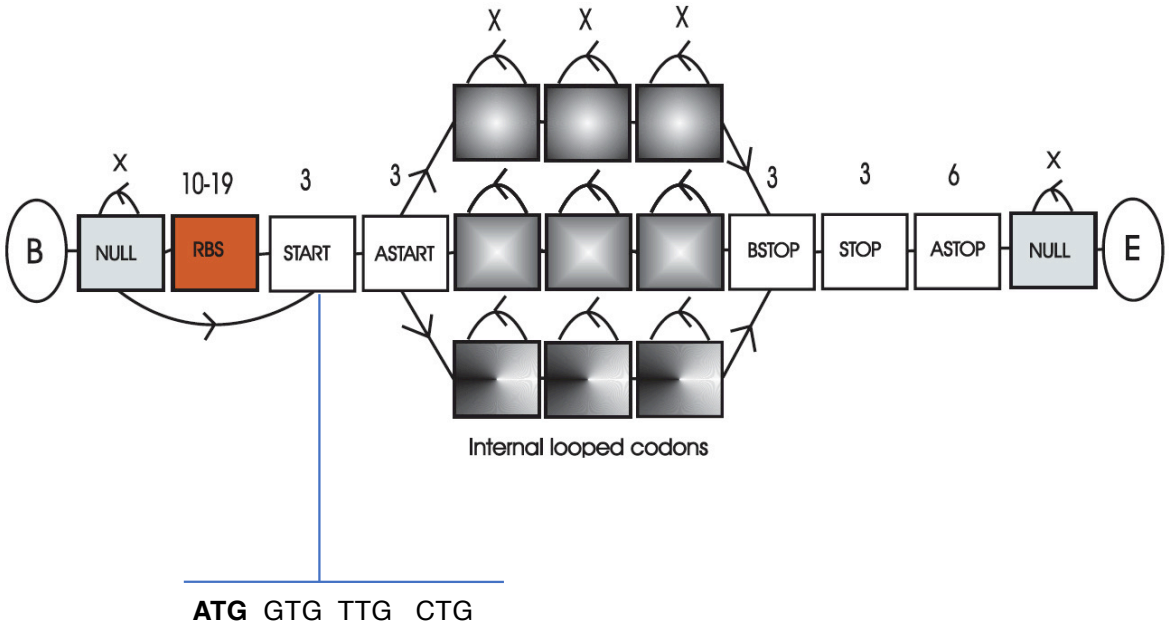
Background

RBS state

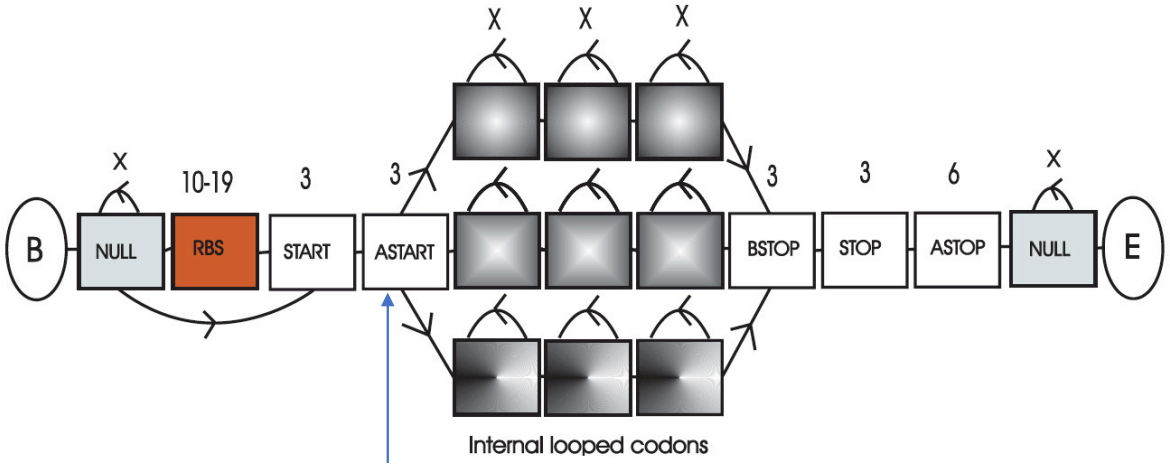


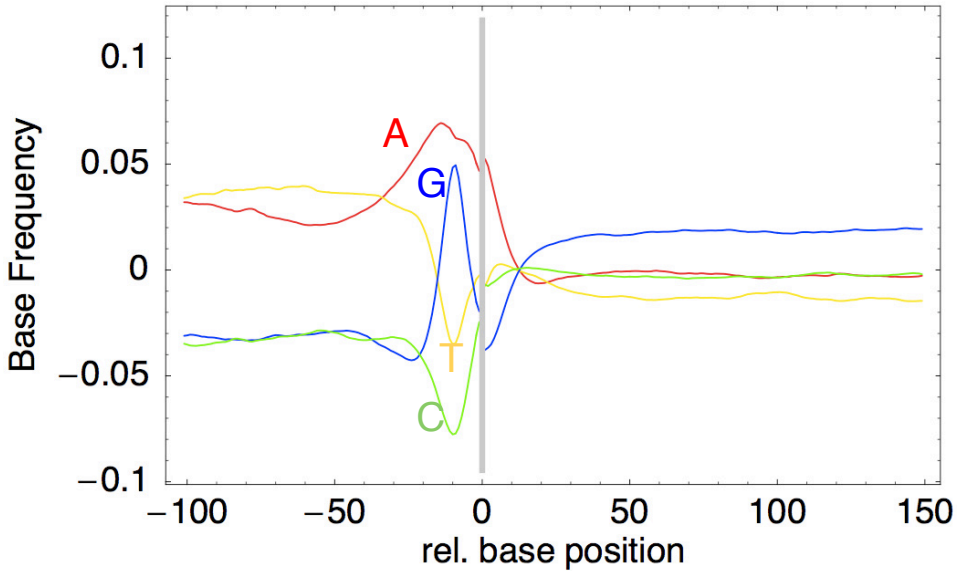
3-12 nucleotides spacer

START state


X
X
X
X
10-19
3
3
3
3
6
X
B
NULL
RBS
START
ASTART
BSTOP
STOP
ASTOP
NULL
E
Internal looped codons
ATG GTG TTG CTG

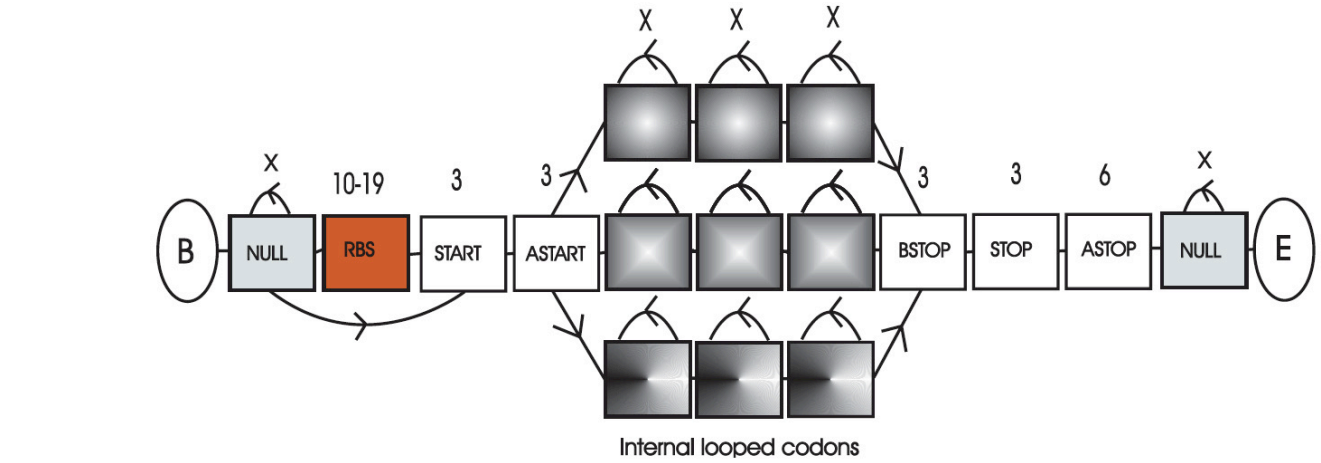
ASTART state





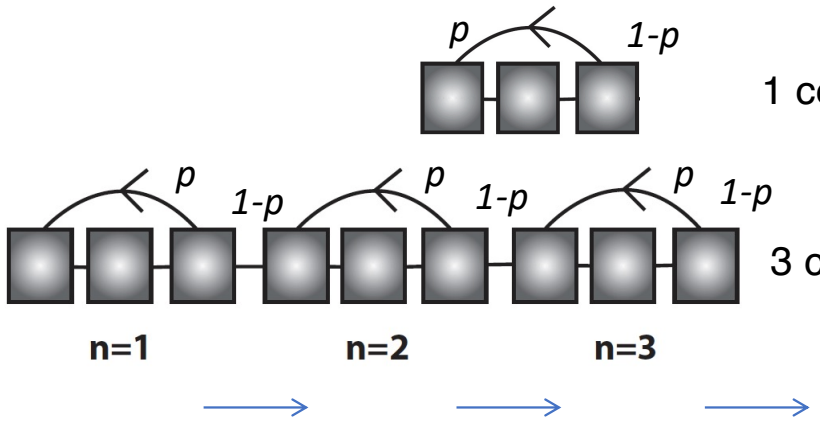
Molina & Van Nimwegen
Genome Res. 2007

Internal looped codons model



Model chosen to get more realistic ORF length distributions than with single codon model.

Internal looped codons model



1 codon model: $P(l+1) = p^l(1-p)$

3 codon model: $P(l+3) = \frac{p^l(1-p)^3(l+1)(l+2)}{2}$

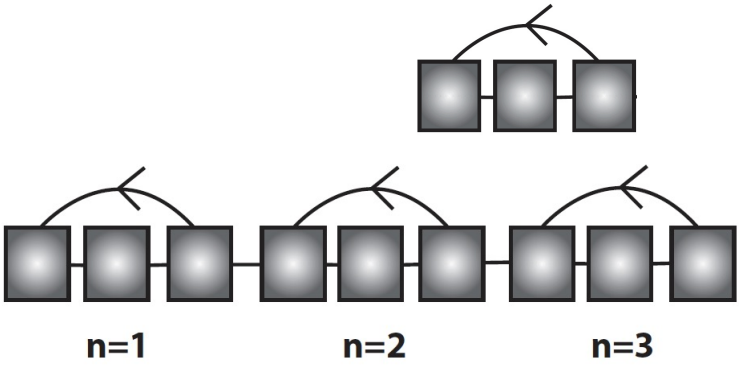
There are 2 transitions to place:

- between codons of type 1 and 2
- between codons of type 2 and 3

None of the states is empty

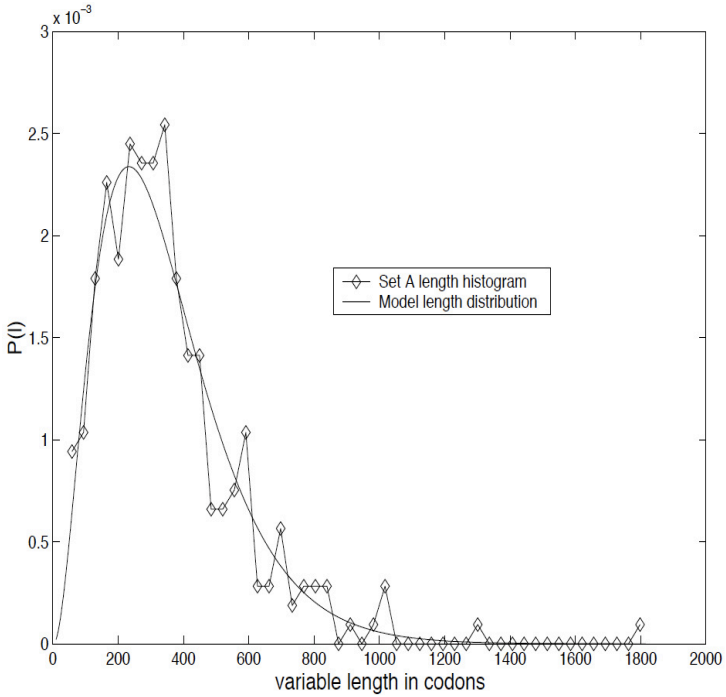
Let's say we have to choose without replacement 2 out of k items in a sequence.
The number of ways we could do this is $(k-1)+(k-2)+\cdots 1 = k(k-1)/2$, where the terms correspond to the number of choices we have for the second item once we pick the first.
In our case $k = l+2$ so the number of ways to place the 2 transitions is $(l+2)(l+1)/2$.

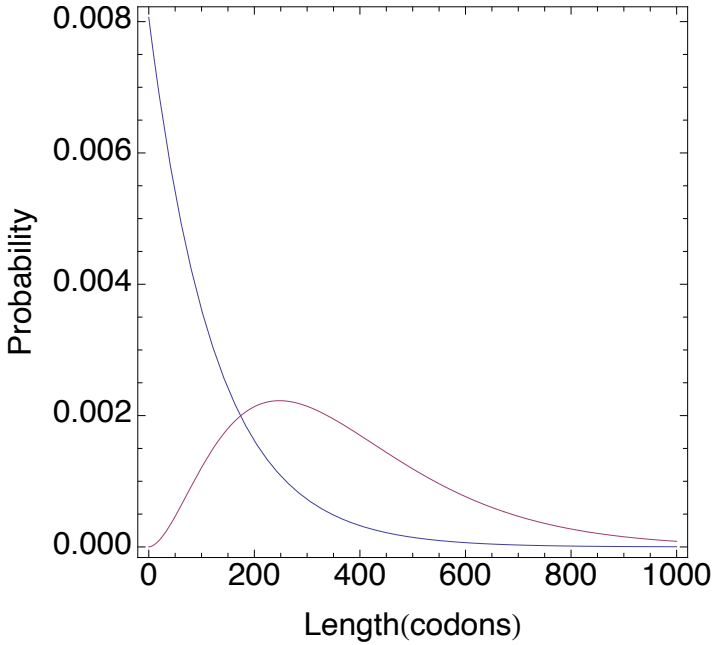
Internal looped codons model



1 codon model: $P(l+1) = p^l(1-p)$

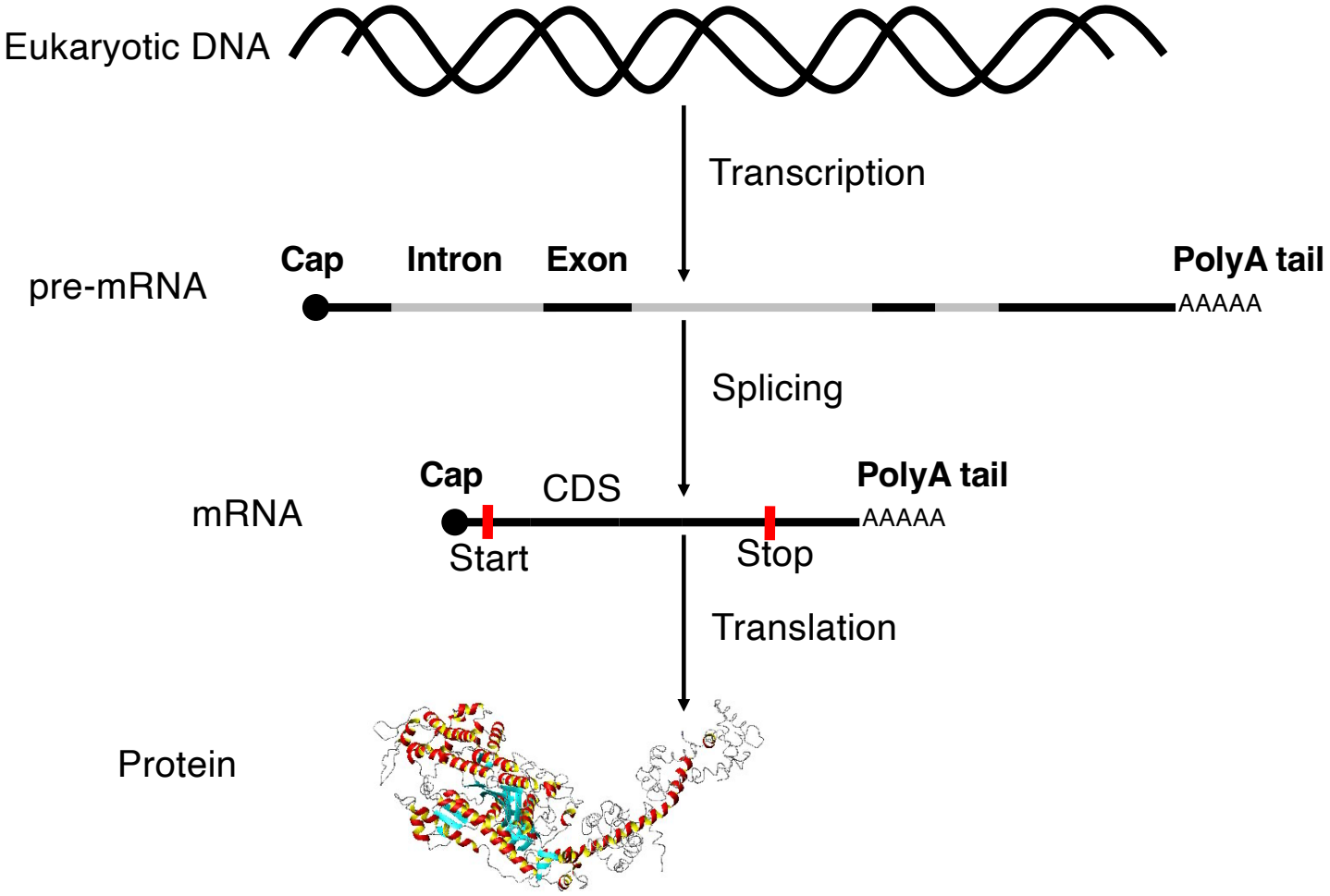
3 codon model: $P(l+3) = \frac{p^l(1-p)^3(l+1)(l+2)}{2}$



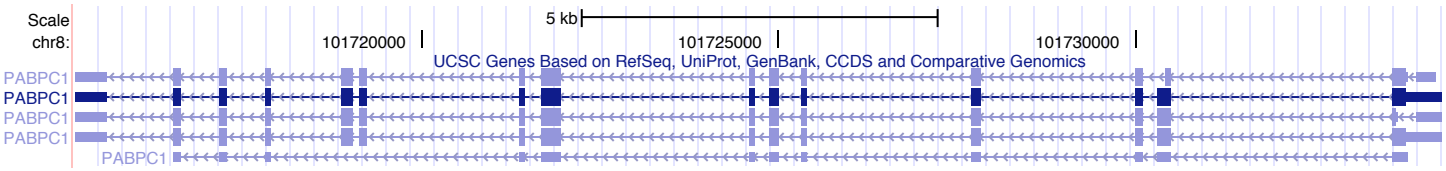


Eukaryotic gene prediction

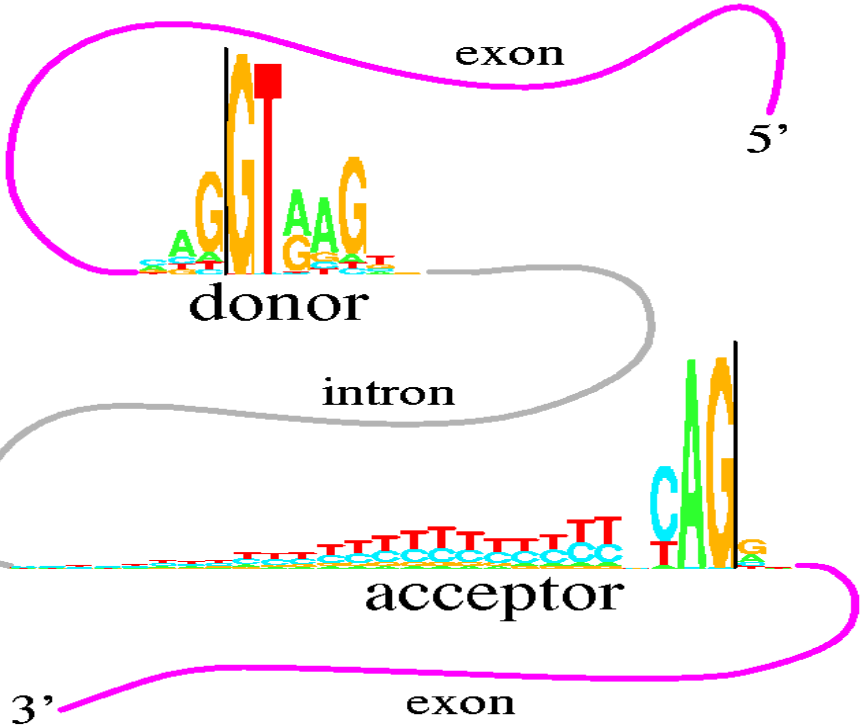
Non-contiguous stretch of nucleotides in the genome that encodes a protein.



Eukaryotic gene structure: split genes



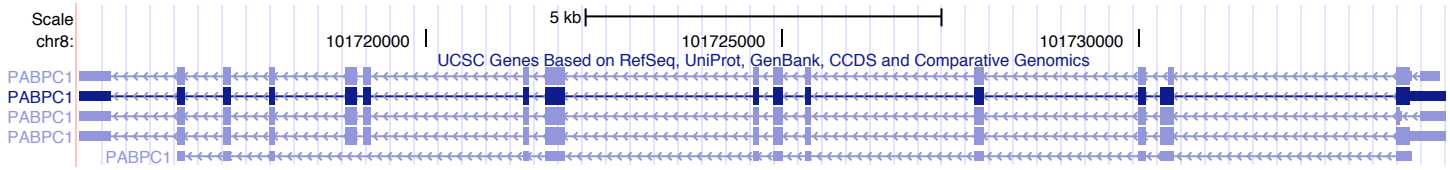
Exons/Introns

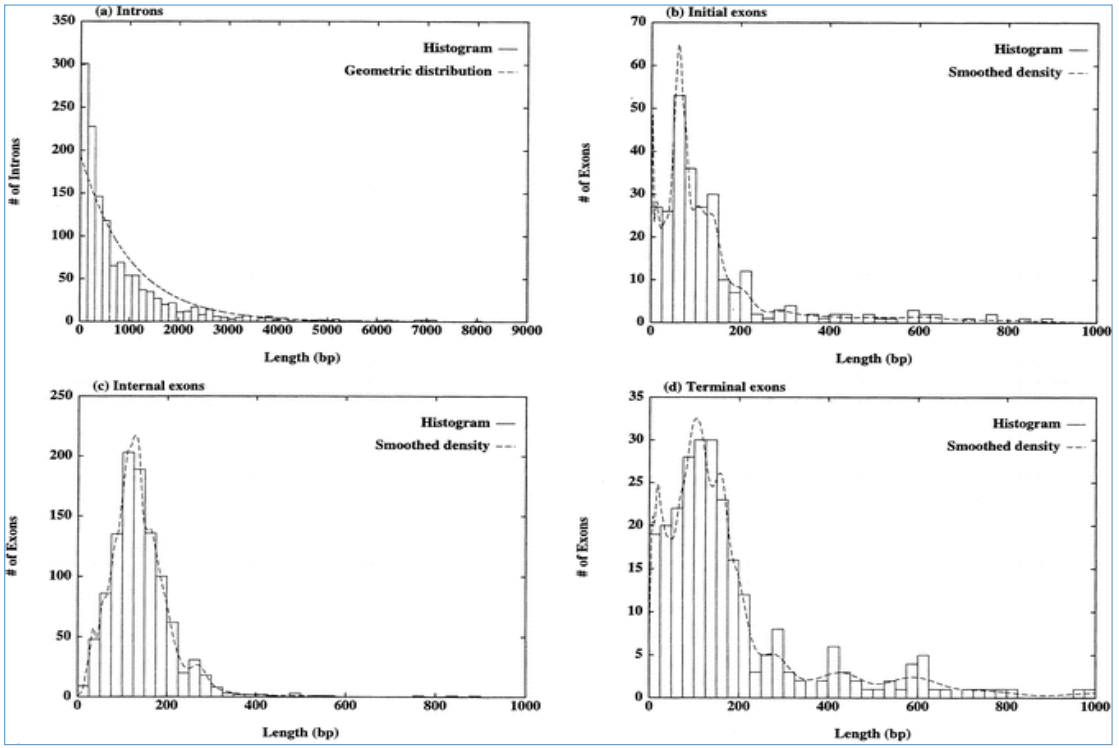


(http://users.fred.net/tds/lab/sequencelogo.html)

tccggcgctgacgctcgcgtaggggccctggcgtcagacgcgcgggggcggggcgagtgcggcgcggggtataagtag
agggtgcaggaggcggtgcttcCCCTTCTCCCCGGCGGTTAGTGCTGAGAGTGCGGAGTGTGTGCTCCGGGCTCGGAA
CACACATTTATTATTAAAAAATCCAAAAAAAATCTAAAAAAATCTTTTAAAAAACCCCAAAAAAATTTACAAAAAATC
CGCGTCTCCCCCGCCGGAGACTTTTATTTTTTTTCTTCCTCTTTTATAAAATAACCCGGTGAAGCAGCCGAGACCGAC
CCGCCCGCCCGCGGCCCCGCAGCAGCTCCAAGAAGGAACCAAGAGACCGAGGCCTTCCCGCTGCCCGGACCCGACACC
GCCACCCTCGCTCCCCGCCGGCAGCCGGCAGCCAGCGGCAGTGGATCGACCCCGTTCTGCGGCCGTTGAgtagttttc
aattccggttgatttttgtccctctgcgcttgctccccgctcccctcccccccggctccggcccccagccccggcactc
gctctcctcctctcacggaaaggtcgcggcctgtggccctgcgggcagccgtgccgagatgaaccccagtgcccccag
ctaccccatggcctcgctctacgtgggggacctccaccccgacgtgaccgaggcgatgctctacgagaagttcagccc
ggccgggcccatcctctccatccgggtctgcagGGACATGATCACCCGCCGCTCCTTGGGCTACGCGTATGTGAACTT
CCAGCAGCCGGCGGACGgtgagcgccgggctccgcggtcggccgggcgggcgggcggccatgaggctcgggcggcggc
ggcgggcggggtaggccggcgggcccggggagggggggcgggaattaattattagcaatatggaattcttttatatgtg
agaaatttcgggggcttctggtatttgtctttgctttatacaacttaggtcccacactagtgtgtgatacttgaaagg
tctagccatcttgaagcttaaagaggaagcgttcaactgtgagaccatgtgtcctcccactccagtcaccatagttta
tataatcttaagtagccagcaaaaattttgaagtaatattgactctgaaaatgtccttcctgcagCGGAGCGTGCTTT
GGACACCATGAATTTTGATGTTATAAAGGGCAAGCCAGTACGCATCATGTGGTCTCAGCGTGATCCATCACTTCGCAA
AAGTGGAGTAGGCAACATATTCATTAAAAATCTGGACAAATCCATTGATAATAAAGCACTGTATGATACATTTTCTGC
TTTTGGTAACATCCTTTCATGTAAGgtaagccaaaaatgtccgatacacatgtttacattgttgagctacatttagtt
tagtgcttgaaataggaactgtgcagtaatggatatctttatttcacaaatatccaggttagatggctaacagattgt
ctctcaacactgaacaaatgttaaatcaatagcaaataaagaaaaatgggtatatgtttttccctagGTGGTTTGTGA
TGAAAATGGTTCCAAGGGCTATGGATTTGTACACTTTGAGACGCAGGAAGCAGCTGAAAGAGCTATTGAAAAAATGAA
TGGAATGCTCCTAAATGATCGCAAAGTgtaagttataaatatctagctaatcatattttgaatcttatttttaataat
tccaaatccattcatttaagttaatatttgtcagggagaagttagatgcacataagataaaagaataaaaacatttt
aaagatagaggcttattcatatatttcatggtgtcatatcttaaattatggccctatttctattttgtagtgtgcttt
ttattagggattctgaatattctcattaacagtaacaaaaccaaccatcaggcttacttaatctgaactttataaaat
agaatctaatgttatagtccttgtatatcttattagATTTGTTGGACGATTTAAGTCTCGTAAAGAACGAGAAGCTGA
ACTTGGAGCTAGGGCAAAAGAATTCACCAATGTTTACATCAAGAATTTTGGAGAAGACATGGATGATGAGCGCCTTAA
GGATCTCTTTGGCAAGTTTGgtaatgtgtcttaattaaatttttacacacaaagctctgaattagtgctcagtgttaa
ttttagctattactttatactaaaaaagtcaataaaattagacttacatggtactttaaaatattttactacattaag

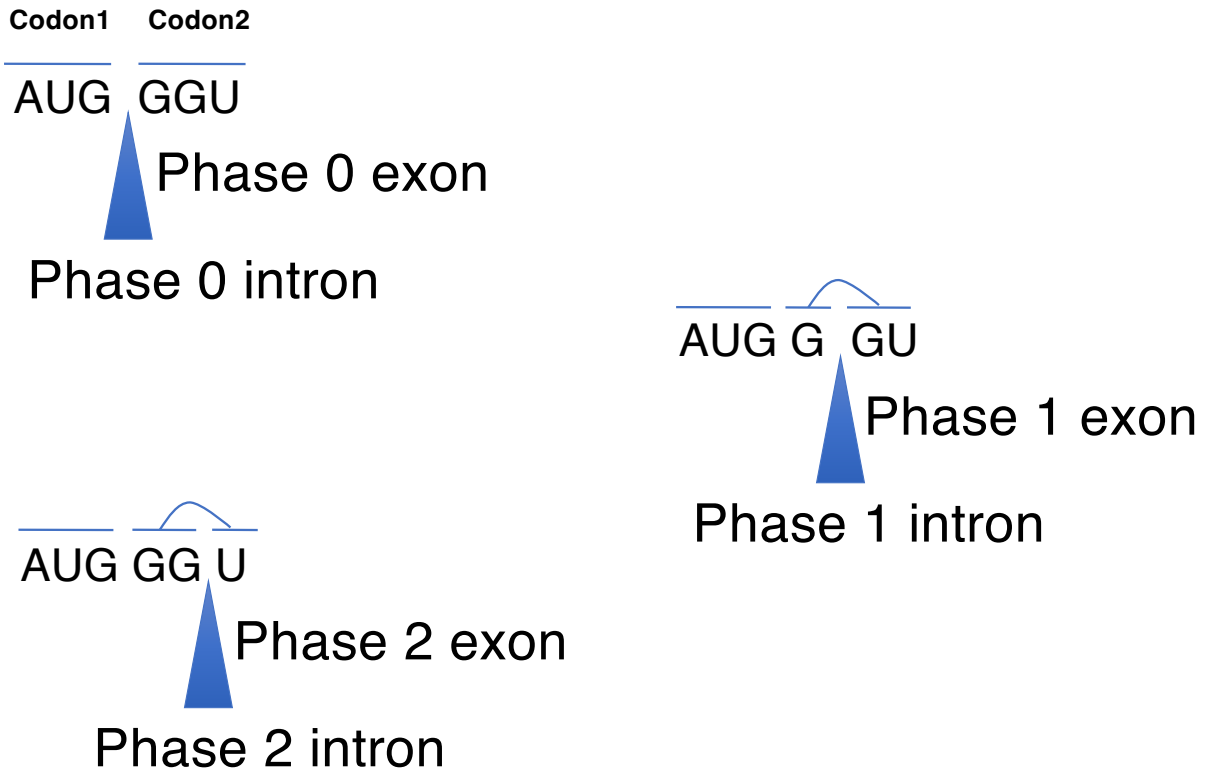
Exon/Intron length distributions



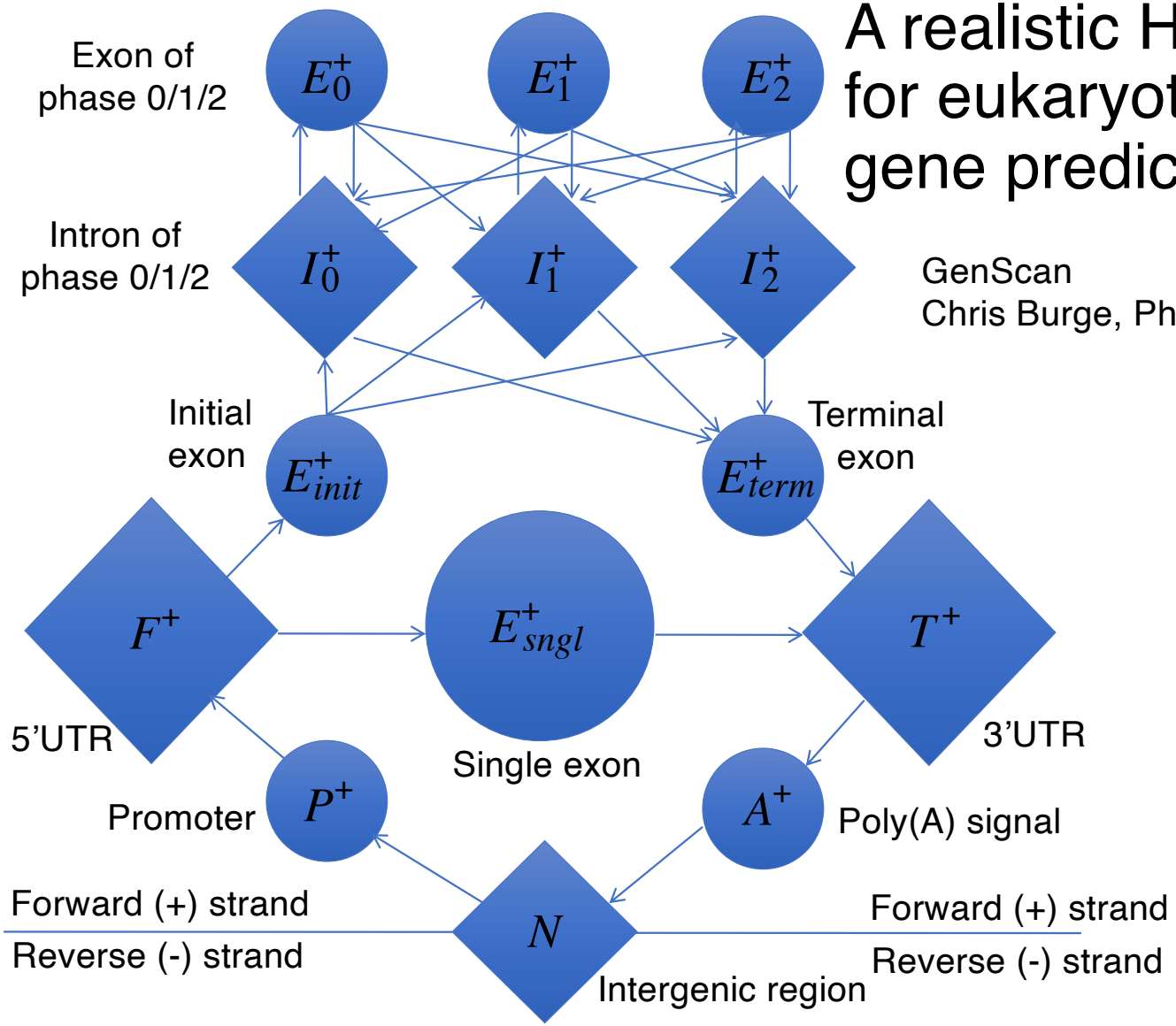



Next intron/exon:
Phase 1
Next intron/exon:
Phase 0

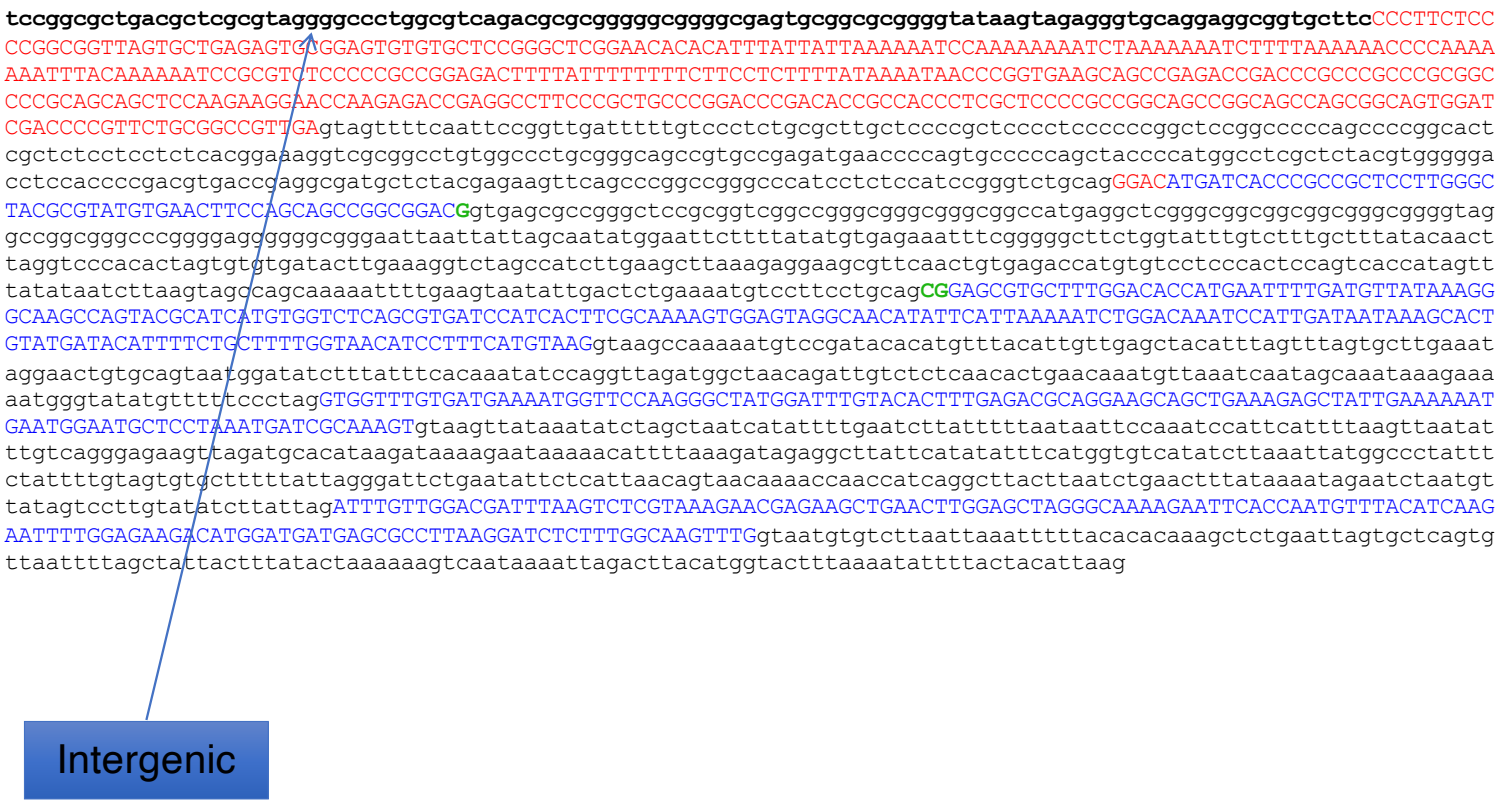
Intron/exon phase



A realistic HMM for eukaryotic gene prediction

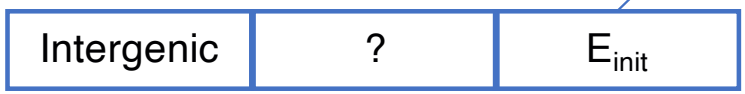


tccggcgctgacgctcgcgtaggggccctggcgtcagacgcgcgggggcggggcgagtgcggcgcggggtataagtagagggtgcaggaggcggtgcttcCCCTTCTCC
CCGGCGGTTAGTGCTGAGAGTGCGGAGTGTGTGCTCCGGGCTCGGAACACACATTTATTATTAAAAAATCCAAAAAAAATCTAAAAAAATCTTTTAAAAAACCCCAAAA
AAATTTACAAAAAATCCGCGTCTCCCCCGCCGGAGACTTTTATTTTTTTTCTTCCTCTTTTATAAAATAACCCGGTGAAGCAGCCGAGACCGACCCGCCCGCCCGCGGC
CCCGCAGCAGCTCCAAGAAGGAACCAAGAGACCGAGGCCTTCCCGCTGCCCGGACCCGACACCGCCACCCTCGCTCCCCGCCGGCAGCCGGCAGCCAGCGGCAGTGGAT
CGACCCCGTTCTGCGGCCGTTGAgtagttttcaattccggttgatttttgtccctctgcgcttgctccccgctcccctcccccggctccggcccccagccccggcact
cgctctcctcctctcacggaaaggtcgcggcctgtggccctgcgggcagccgtgccgagatgaaccccagtgcccccagctaccccatggcctcgctctacgtgggggga
cctccaccccgacgtgaccgaggcgatgctctacgagaagttcagcccggccgggcccatcctctccatccgggtctgcagGGACATGATCACCCGCCGCTCCTTGGGC
TACGCGTATGTGAACTTCCAGCAGCCGGCGGACGgtgagcgccgggctccgcggtcggccgggcgggcgggcggccatgaggctcgggcggcggcggcgggcggggtag
gccggcgggcccggggagggggcgggaattaattattagcaatatggaattcttttatatgtgagaaatttcgggggcttctggtatttgtctttgctttatacaact
taggtcccacactagtgtgtgatacttgaaaggtctagccatcttgaagcttaaagaggaagcgttcaactgtgagaccatgtgtcctcccactccagtcaccatagtt
tatataatcttaagtagccagcaaaaattttgaagtaatattgactctgaaaatgtccttcctgcagCGGAGCGTGCTTTGGACACCATGAATTTTGATGTTATAAAGG
GCAAGCCAGTACGCATCATGTGGTCTCAGCGTGATCCATCACTTCGCAAAAGTGGAGTAGGCAACATATTCATTAAAAATCTGGACAAATCCATTGATAATAAAGCACT
GTATGATACATTTTCTGCTTTTGGTAACATCCTTTCATGTAAGgtaagccaaaaatgtccgatacacatgtttacattgttgagctacatttagtttagtgcttgaaat
aggaactgtgcagtaatggatatctttatttcacaaatatccaggttagatggctaacagattgtctctcaacactgaacaaatgttaaatcaatagcaaataaagaaa
aatgggtatatgtttttccctagGTGGTTTGTGATGAAAATGGTTCCAAGGGCTATGGATTTGTACACTTTGAGACGCAGGAAGCAGCTGAAAGAGCTATTGAAAAAAT
GAATGGAATGCTCCTAAATGATCGCAAAGTgtaagttataaatatctagctaatcatattttgaatcttatttttaataattccaaatccattcattttaagttaatat
ttgtcagggagaagttagatgcacataagataaaagaataaaaacattttaaagatagaggcttattcatatatttcatggtgtcatatcttaaattatggccctattt
ctattttgtagtgtgctttttattagggattctgaatattctcattaacagtaacaaaaccaaccatcaggcttacttaatctgaacttttataaaatagaatctaatgt
tatagtccttgtatatcttattagATTTGTTGGACGATTTAAGTCTCGTAAAGAACGAGAAGCTGAACTTGGAGCTAGGGCAAAAGAATTCACCAATGTTTACATCAAG
AATTTTGGAGAAGACATGGATGATGAGCGCCTTAAGGATCTCTTTGGCAAGTTTGgtaatgtgtcttaattaaatttttacacacaaagctctgaattagtgctcagtg
ttaattttagctattactttatactaaaaaagtcaataaaattagacttacatggtactttaaaatattttactacattaag

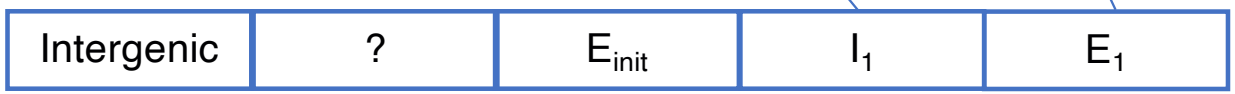

tccggcgctgacgctcgcgtaggggccctggcgtcagacgcgcgggggcggggcgagtgcggcgcggggtataagtagagggtgcaggaggcggtgcttcCCCTTCTCC
CCGGCGGTTAGTGCTGAGAGTGTGGAGTGTGTGCTCCGGGCTCGGAACACACATTTATTATTAAAAAATCCAAAAAAAATCTAAAAAAATCTTTTAAAAAACCCCAAAA
AAATTTACAAAAAATCCGCGTCTCCCCCGCCGGAGACTTTTATTTTTTTTCTTCCTCTTTTATAAAATAACCCGGTGAAGCAGCCGAGACCGACCCGCCCGCCCGCGGC
CCCGCAGCAGCTCCAAGAAGGAACCAAGAGACCGAGGCCTTCCCGCTGCCCGGACCCGACACCGCCACCCTCGCTCCCCGCCGGCAGCCGGCAGCCAGCGGCAGTGGAT
CGACCCCGTTCTGCGGCCGTTGAgtagttttcaattccggttgatttttgtccctctgcgcttgctccccgctcccctcccccggctccggcccccagccccggcact
cgctctcctcctctcacggaaaggtcgcggcctgtggccctgcgggcagccgtgccgagatgaaccccagtgcccccagctaccccatggcctcgctctacgtgggggga
cctccaccccgacgtgaccgaggcgatgctctacgagaagttcagcccggccgggcccatcctctccatccgggtctgcagGGACATGATCACCCGCCGCTCCTTGGGC
TACGCGTATGTGAACTTCCAGCAGCCGGCGGACGgtgagcgccgggctccgcggtcggccgggcgggcgggcggccatgaggctcgggcggcggcggcgggcggggtag
gccggcgggcccggggaggggggcgggaattaattattagcaatatggaattcttttatatgtgagaaatttcgggggcttctggtatttgtctttgctttatacaact
taggtcccacactagtgtgtgatacttgaaaggtctagccatcttgaagcttaaagaggaagcgttcaactgtgagaccatgtgtcctcccactccagtcaccatagtt
tatataatcttaagtagccagcaaaaattttgaagtaatattgactctgaaaatgtccttcctgcagCGGAGCGTGCTTTGGACACCATGAATTTTGATGTTATAAAGG
GCAAGCCAGTACGCATCATGTGGTCTCAGCGTGATCCATCACTTCGCAAAAGTGGAGTAGGCAACATATTCATTAAAAATCTGGACAAATCCATTGATAATAAAGCACT
GTATGATACATTTTCTGCTTTTGGTAACATCCTTTCATGTAAGgtaagccaaaaatgtccgatacacatgtttacattgttgagctacatttagtttagtgcttgaaat
aggaactgtgcagtaatggatatctttatttcacaaatatccaggttagatggctaacagattgtctctcaacactgaacaaatgttaaatcaatagcaaataaagaaa
aatgggtatatgttttccctagGTGGTTTGTGATGAAAATGGTTCCAAGGGCTATGGATTTGTACACTTTGAGACGCAGGAAGCAGCTGAAAGAGCTATTGAAAAAAT
GAATGGAATGCTCCTAAATGATCGCAAAGTgtaagttataaatatctagctaatcatattttgaatcttatttttaataattccaaatccattcattttaagttaatat
ttgtcagggagaagttagatgcacataagataaaagaataaaaacattttaaagatagaggcttattcatatatttcatggtgtcatatcttaaattatggccctattt
ctattttgtagtgtgcttttttattagggattctgaatattctcattaacagtaacaaaaccaaccatcaggcttacttaatctgaacttttataaaatagaatctaatgt
tatagtccttgtatatcttattagATTTGTTGGACGATTTAAGTCTCGTAAAGAACGAGAAGCTGAACTTGGAGCTAGGGCAAAAGAATTCACCAATGTTTACATCAAG
AATTTTGGAGAAGACATGGATGATGAGCGCCTTAAGGATCTCTTTGGCAAGTTTGgtaatgtgtcttaattaaatttttacacacaaagctctgaattagtgctcagtg
ttaattttagctacttactttatactaaaaaagtcaataaaattagacttacatggtactttaaaatattttactacattaag
Intergenic

tccggcgctgacgctcgcgtaggggccctggcgtcagacgcgcgggggcggggcgagtgcggcgcggggtataagtagagggtgcaggaggcggtgcttc**CCCTTCTCC**
CCGGCGGTTAGTGCTGAGAGTGCGGAGTGTGTGCTCCGGGCTCGGAACACACATTTATTATTAAAAAATCCAAAAAAAATCTAAAAAAATCTTTTAAAAAACCCCAAAA
AAATTTACAAAAAATCCGCGTCTCCCCCGCCGGAGACTTTTATTTTTTTTCTTCCTCTTTTATAAAATAACCCGGTGAAGCAGCCGAGACCGACCCGCCCGCCCGCGGC
CCCGCAGCAGCTCCAAGAAGGAACCAAGAGACCGAGGCCTTCCCGCTGCCCGGACCCGACACCGCCACCCTCGCTCCCCGCCGGCAGCCGGCAGCCAGCGGCAGTGGAT
CGACCCCGTTCTGCGGCCGTTGAgtagttttcaattccggttgatttttgtccctctgccgcttgctccccgctcccctcccccggctccggcccccagccccggcact
cgctctcctcctctcacggaaaggtcgcggcctgtggccctgcgggcagccgtgccgagatgaaccccagtgcccccagctaccccatggcctcgctctacgtgggggga
cctccaccccgacgtgaccgaggcgatgctctacgagaagttcagcccggccgggcccatcctctccatccgggtctgcag**GGAC**ATGATCACCCGCCGCTCCTTGGGC
TACGCGTATGTGAACTTCCAGCAGCCGGCGGAC**G**gtgagcgccgggctccgcggtcggccgggcgggcgggcggccatgaggctcgggcggcggcggcgggcggggtag
gccggcgggcccggggagggggcgggaattaattattagcaatatggaattctttatatgtgagaaatttcgggggcttctggtatttgtctttgctttatacaact
taggtcccacactagtgtgtgatacttgaaaggtctagccatcttgaagcttaaagaggaagcgttcaactgtgagaccatgtgtcctcccactccagtcaccatagtt
tatataatcttaagtagccagcaaaaattttgaagtaatattgactctgaaaatgtccttcctgcag**CG**GAGCGTGCTTTGGACACCATGAATTTTGATGTTATAAAGG
GCAAGCCAGTACGCATCATGTGGTCTCAGCGTGATCCATCACTTCGCAAAAGTGGAGTAGGCAACATATTCATTAAAAATCTGGACAAATCCATTGATAATAAAGCACT
GTATGATACATTTTCTGCTTTTGGTAACATCCTTTCATGTAAGgtaagccaaaaatgtccgatacacatgtttacattgttgagctacatttagtttagtgcttgaaat
aggaactgtgcagtaatggatatctttatttcacaaatatccaggttagatggctaacagattgtctctcaacactgaacaaatgttaaatcaatagcaaataaagaaa
aatgggtatatgtttttccctagGTGGTTTGTGATGAAAATGGTTCCAAGGGCTATGGATTTGTACACTTTGAGACGCAGGAAGCAGCTGAAAGAGCTATTGAAAAAAT
GAATGGAATGCTCCTAAATGATCGCAAAGTgtaagttataaatatctagctaatcatatttttgaatcttatttttaataattccaaatccattcattttaagttaatat
ttgtcagggagaagttagatgcacataagataaaagaataaaacatttttaaagatagaggcttattcatatatttcatggtgtcatatcttaaattatggccctattt
ctattttgtagtgtgctttttattagggattctgaatattcttcattaacagtaacaaaaccaaccatcaggcttacttaatctgaacttttataaaatagaatctaatgt
tatagtccttgtatatcttattagATTTGTTGGACGATTTAAGTCTCGTAAAGAACGAGAAGCTGAACTTGGAGCTAGGGCAAAAGAATTCACCAATGTTTACATCAAG
AATTTTGGAGAAGACATGGATGATGAGCGCCTTAAGGATCTCTTTGGCAAGTTTGgtaatgtgtcttaattaaatttttacacacaaagctctgaattagtgctcagtg
ttaattttagctattactttatactaaaaaagtcaataaaattagacttacatggtactttaaaatattttactacattaag

tccggcgctgacgctcgcgtaggggccctggcgtcagacgcgcgggggcggggcgagtgcggcgcggggtataagtagagggtgcaggaggcggtgcttcCCCTTCTCC
CCGGCGGTTAGTGCTGAGAGTGCGGAGTGTGTGCTCCGGGCTCGGAACACACATTTATTATTAAAAAATCCAAAAAAAATCTAAAAAAATCTTTTAAAAAACCCCAAAA
AAATTTACAAAAAATCCGCGTCTCCCCCGCCGGAGACTTTTATTTTTTTTCTTCCTCTTTTATAAAATAACCCGGTGAAGCAGCCGAGACCGACCCGCCCGCCCGCGGC
CCCGCAGCAGCTCCAAGAAGGAACCAAGAGACCGAGGCCTTCCCGCTGCCCGGACCCGACACCGCCACCCTCGCTCCCCGCCGGCAGCCGGCAGCCAGCGGCAGTGGAT
CGACCCCGTTCTGCGGCCGTTGA**gtagttttcaattccggttgatttttgtccctctgcgcttgctccccgctcccctcccccggctccggcccccagccccggcact**
cgctctcctcctctcacggaaaggtcgcggcctgtggccctgcgggcagccgtgccgagatgaaccccagtgcccccagctaccccatggcctcgctctacgtgggggga
cctccaccccgacgtgaccgaggcgatgctctacgagaagttcagcccggccgggcccatcctctccatccgggtctgcagGGACATGATCACCCGCCGCTCCTTGGGC
TACGCGTATGTGAACTTCCAGCAGCCGGCGGACGgtgagcgccgggctccgcggtcggccgggcgggcgggcggccatgaggctcgggcggcggcggcgggcggggtag
gccggcgggcccggggagggggcgggaattaattattagcaatatggaattcttttatatgtgagaaatttcgggggcttctggtatttgtctttgctttatacaact
taggtcccacactagtgtgtgatacttgaaaggtctagccatcttgaagcttaaagaggaagcgttcaactgtgagaccatgtgtcctcccactccagtcaccatagtt
tatataatcttaagtagccagcaaaaattttgaagtaatattgactctgaaaatgtccttcctgcagCGGAGCGTGCTTTGGACACCATGAATTTTGATGTTATAAAGG
GCAAGCCAGTACGCATCATGTGGTCTCAGCGTGATCCATCACTTCGCAAAAGTGGAGTAGGCAACATATTCATTAAAAATCTGGACAAATCCATTGATAATAAAGCACT
GTATGATACATTTTCTGCTTTTGGTAACATCCTTTCATGTAAGgtaagccaaaaatgtccgatacacatgtttacattgttgagctacatttagtttagtgcttgaaat
aggaactgtgcagtaatggatatctttatttcacaaatatccaggttagatggctaacagattgtctctcaacactgaacaaatgttaaatcaatagcaaataaagaaa
aatgggtatatgtttttccctagGTGGTTTGTGATGAAAATGGTTCCAAGGGCTATGGATTTGTACACTTTGAGACGCAGGAAGCAGCTGAAAGAGCTATTGAAAAAAT
GAATGGAATGCTCCTAAATGATCGCAAAGTgtaagttataaatatctagctaatcatattttgaatcttatttttaataattccaaatccattcattttaagttaatat
ttgtcagggagaagttagatgcacataagataaaagaataaaaacattttaaagatagaggcttattcatatatttcatggtgtcatatcttaaattatggccctattt
ctattttgtagtgtgctttttattagggattctgaatattctcattaacagtaacaaaaccaaccatcaggcttacttaatctgaacttttataaaatagaatctaatgt
tatagtccttgtatatcttattagATTTGTTGGACGATTTAAGTCTCGTAAAGAACGAGAAGCTGAACTTGGAGCTAGGGCAAAAGAATTCACCAATGTTTACATCAAG
AATTTTGGAGAAGACATGGATGATGAGCGCCTTAAGGATCTCTTTGGCAAGTTTGgtaatgtgtcttaattaaatttttacacacaaagctctgaattagtgctcagtg
ttaattttagctattactttatactaaaaaagtcaataaaattagacttacatggtactttaaaatattttactacattaag



tccggcgctgacgctcgcgtaggggccctggcgtcagacgcgcgggggcggggcgagtgcggcgcggggtataagtagagggtgcaggaggcggtgcttcCCCTTCTCC
CCGGCGGTTAGTGCTGAGAGTGCGGAGTGTGTGCTCCGGGCTCGGAACACACATTTATTATTAAAAAATCCAAAAAAAATCTAAAAAAATCTTTTAAAAAACCCCAAAA
AAATTTACAAAAAATCCGCGTCTCCCCCGCCGGAGACTTTTATTTTTTTTCTTCCTCTTTTATAAAATAACCCGGTGAAGCAGCCGAGACCGACCCGCCCGCCCGCGGC
CCCGCAGCAGCTCCAAGAAGGAACCAAGAGACCGAGGCCTTCCCGCTGCCCGGACCCGACACCGCCACCCTCGCTCCCCGCCGGCAGCCGGCAGCCAGCGGCAGTGGAT
CGACCCCGTTCTGCGGCCGTTGAgtagttttcaattccggttgatttttgtccctctgcgcttgctccccgctcccctcccccggctccggcccccagccccggcact
cgctctcctcctctcacggaaaggtcgcggcctgtggccctgcgggcagccgtgccgagatgaaccccagtgcccccagctaccccatggcctcgctctacgtgggggga
cctccaccccgacgtgaccgaggcgatgctctacgagaagttcagcccggccgggcccatcctctccatccgggtctgcagGGACATGATCACCCGCCGCTCCTTGGGC
TACGCGTATGTGAACTTCCAGCAGCCGGCGGACG**gtgagcgccgggctccgcggtcggccgggcgggcgggcggccatgaggctcgggcggcggcggcgggcggggtag**
gccggcgggcccggggagggggcgggaattaattcattagcaatatggaattcttttatatgtgagaaatttcgggggcttctggtatttgtctttgctttatacaact
taggtcccacactagtgtgtgatacttgaaaggtctagccatcttgaagcttaaagaggaagcgttcaactgtgagaccatgtgtcctcccactccagtcaccatagtt
tatataatcttaagtagccagcaaaaattttgaagtaatattgactctgaaaatgtccttcctgcagCGGAGCGTGCTTTGGACACCATGAATTTTGATGTTATAAAGG
GCAAGCCAGTACGCATCATGTGGTCTCAGCGTGATCCATCACTTCGCAAAAGTGGAGTAGGCAACATATTCATTAAAAATCTGGACAAATCCATTGATAATAAAGCACT
GTATGATACATTTTCTGCTTTTGGTAACATCCTTTCATGTAAGgtaagccaaaaatgtccgatacacatgtttacattgttgagctacatttagtttagtgcttgaaat
aggaactgtgcagtaatggatatctttatttcacaaatatccaggttagatggctaacagattgtctctcaacactgaacaaatgttaaatcaatagcaaataaagaaa
aatgggtatatgtttttccctagGTGGTTTGTGATGAAAATGGTTCCAAGGGCTATGGATTTGTACACTTTGAGACGCAGGAAGCAGCTGAAAGAGCTATTGAAAAAAT
GAATGGAATGCTCCTAAATGATCGCAAAGTgtaagttataaatatctagctaatcatattttgaatcttatttttaataattccaaatccattcattttaagttaatat
ttgtcagggagaagttagatgcacataagataaaagaataaaaacattttaaagatagaggcttattcatatatttcatggtgtcatatcttaaattatggccctattt
ctattttgtagtgtgctttttattagggattctgaatattctcattaacagtaacaaaaccaaccatcaggcttacttaatctgaacttttataaaatagaatctaatgt
tatagtccttgtatatcttattagATTTGTTGGACGATTTAAGTCTCGTAAAGAACGAGAAGCTGAACTTGGAGCTAGGGCAAAAGAATTCACCAATGTTTACATCAAG
AATTTTGGAGAAGACATGGATGATGAGCGCCTTAAGGATCTCTTTGGCAAGTTTGgtaatgtgtcttaattaaatttttacacacaaagctctgaattagtgctcagtg
ttaattttagctattactttatactaaaaaagtcaataaaattagacttacatggtactttaaaatattttactacattaag



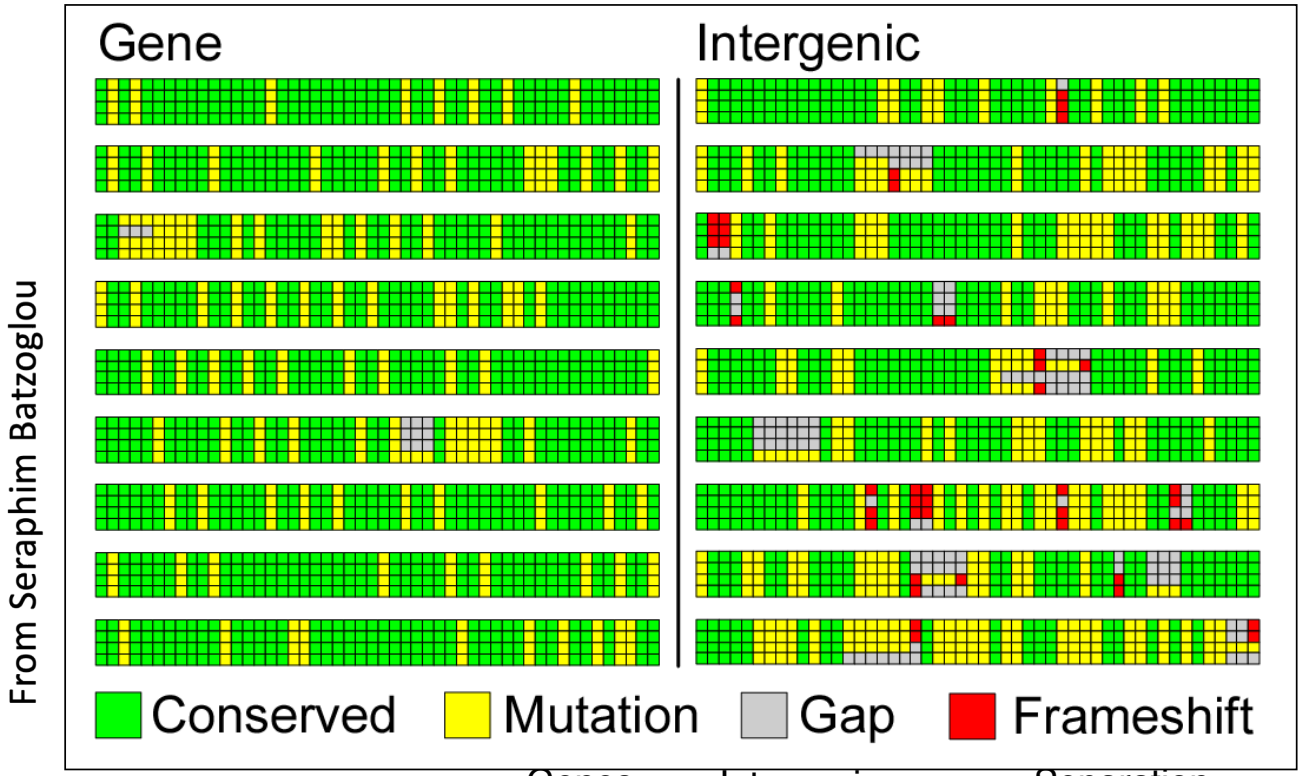
$$P(E_1) \propto P(\text{length} = |\text{E}_1|)P(\text{sequence})$$

$$P(\text{sequence}) \propto P(3'\text{SS})P(\text{exon sequence})P(5'\text{SS})$$

Other features that can be incorporated in gene finders

- Expression data (mRNAs/ESTs).
- Comparative genomic information (gene prediction on pairwise-aligned genome sequences).

Pattern of mutations in different genomic regions



	Genes	Intergenic		Separation
Mutations	30%	58%	→	2-fold
Gaps	1.3%	14%	→	10-fold
Frameshifts	0.14%	10.2%	→	75-fold

Gene finders

Bacterial

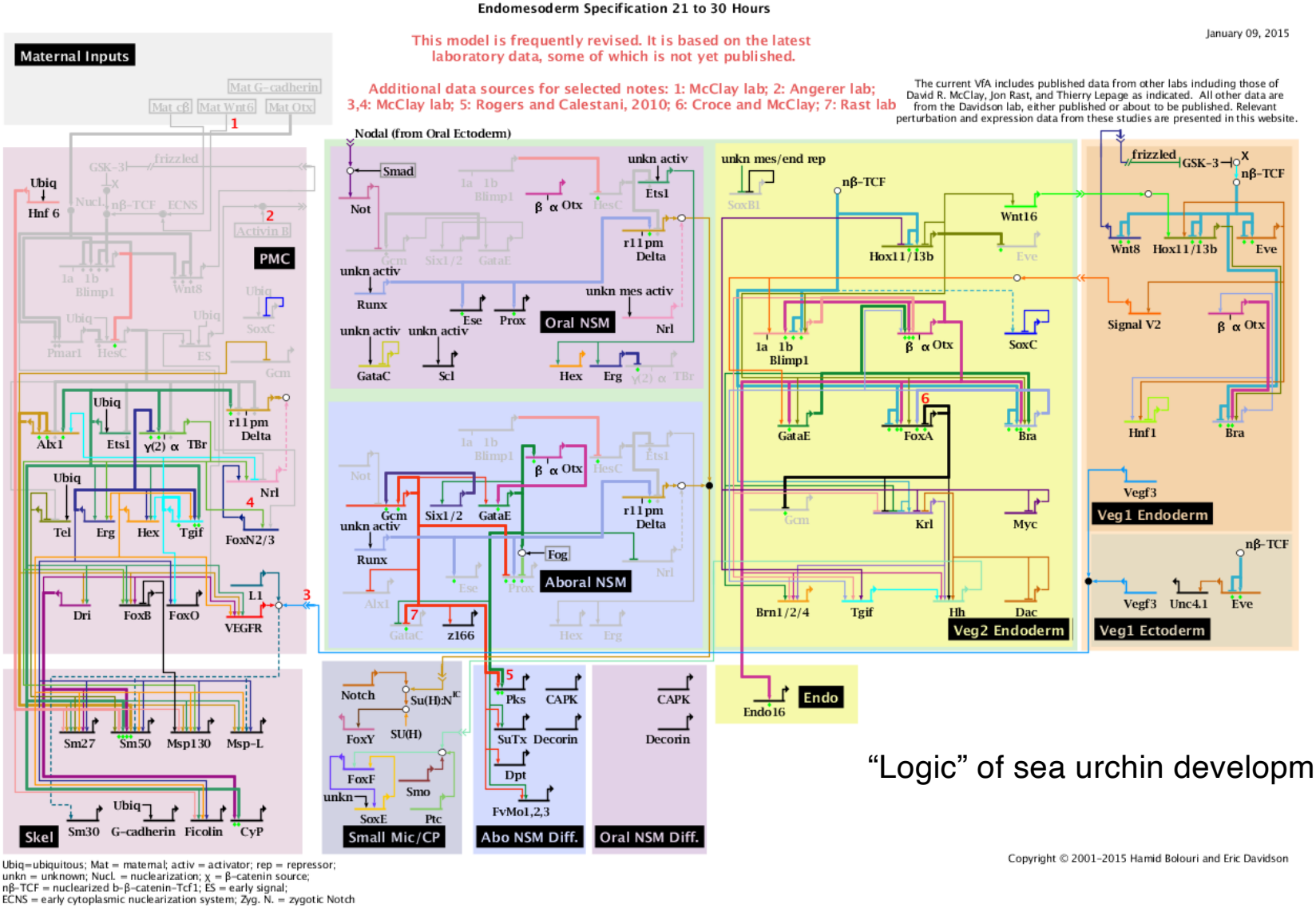
- Glimmer http://ccb.jhu.edu/software/glimmer/index.shtml
- GeneMark http://exon.gatech.edu/GeneMark/
- EasyGene http://www.cbs.dtu.dk/services/EasyGene/

Eukaryotic

- GenScan http://hollywood.mit.edu/GENSCAN.html
- NScan http://mblab.wustl.edu/software/download/
- CONTRAST http://contra.stanford.edu/contrast/

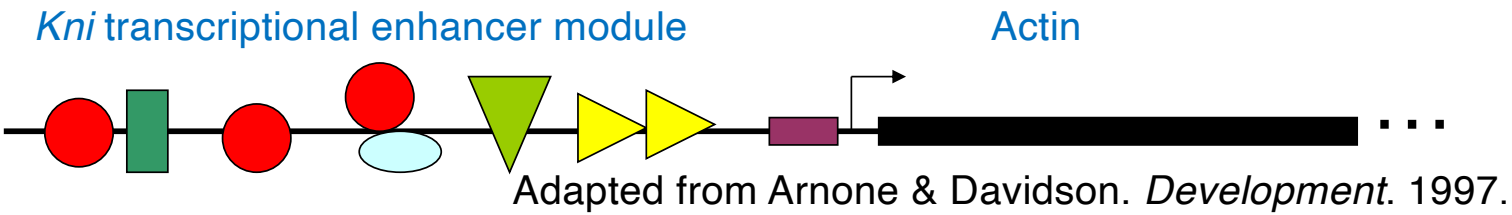
Finding regulatory elements in DNA and RNA

Why?



"Logic" of sea urchin development

RNA synthesis is regulated by transcription factors

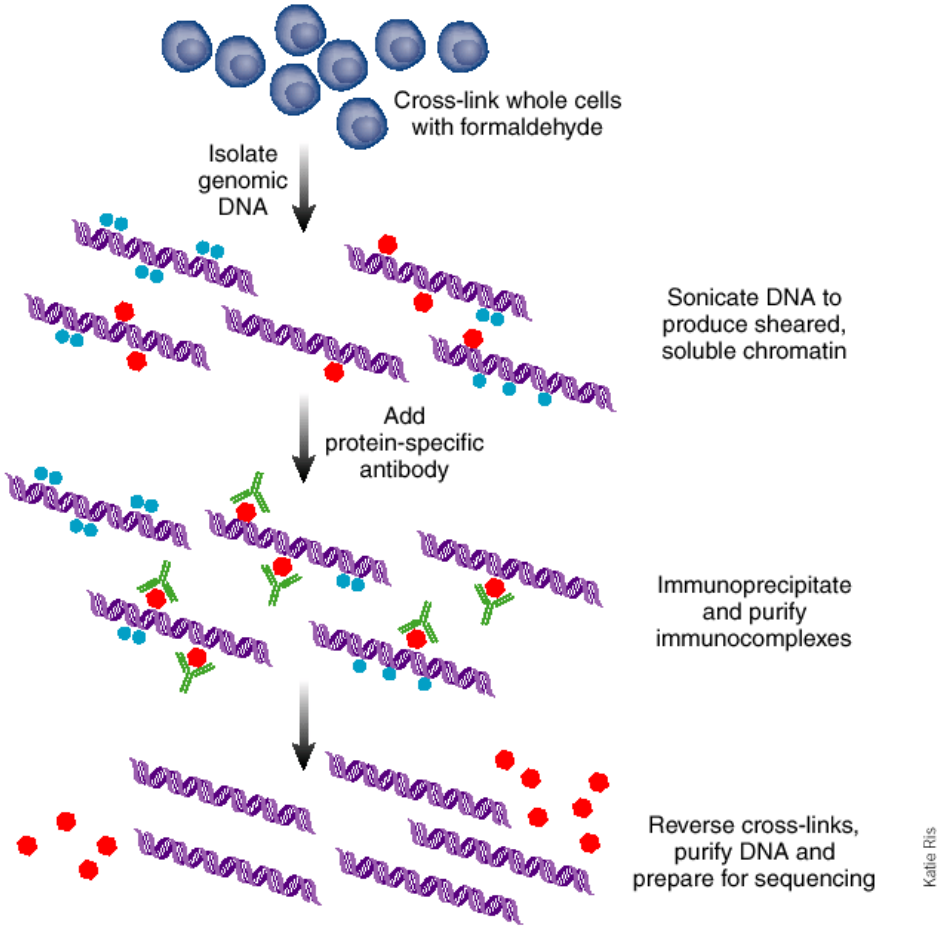


Adapted from Arnone & Davidson. *Development*. 1997.

How do we learn about transcription regulation?

- Which transcription factors regulate the expression of any given gene?
- Where do these transcription factors bind on the DNA?
- What sequence motifs do the transcription factors recognize?

Transcription factor binding sites are identified experimentally by



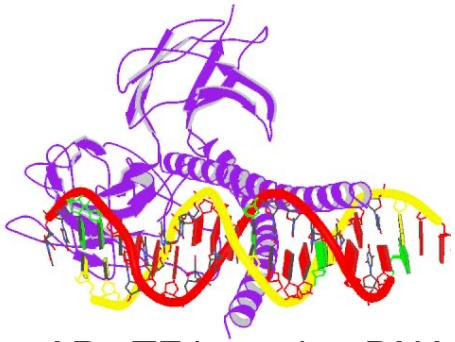
ChIP-seq: chromatin immunoprecipitation and sequencing

Mardis ER. *Nature Methods* **4**, 613 - 614 (2007)

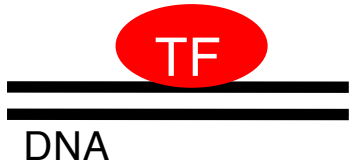
Analysis of ChIP-seq data: computational challenges

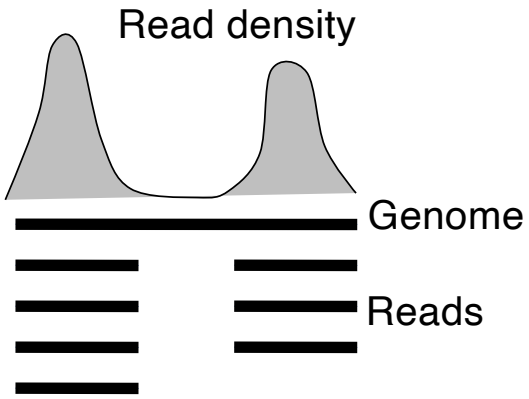
AGTAGTGTGTGCCCGTCTGTTCGTATCTTCTTGCGCTTCG
TGTGTGACTCTGGTAGCTACAAAACAAAAGTCAGCAAAAA
AGATCCCTCAGACCCTTTGTTCGTCGTATGCCGTCTTCTG
GGTAGTGTGGAAAATCTCTAGCAGTGTCGTCGTATGCCGT
GCGCCCGAACAGGGACTTAAAAGCGTCGCTGACCGAATTC
AAAGTAAGACCAGAGGAGATCTCTCGACGCAGGTCGTATG
ACTCGGCTTGCTGAAGTGCACTCGGTCGTCGTATGCCGTC
AGGGGCGGCTGGTGAGTACGCCATTTTTTCGGATGAAGTC
TGGGTGCGAGAGCGTTCGTACGTCTGCTAATGCTTCAATA
TAAGGCCAGGGTCGTATGCCGTCTTCTGCTTGGAAAGAAA
TGGAAAGATTTGCACTCATCGCGATGCCTCCTCTGCTTAC
TGAAACAGCTACAACCAGCTCTACAGACAGGAACAGAGGA
CAGTAGCAACTCTCTATTGTCGTAAGACGTCAACTGCTTT
TAGACAAGATAGAGGAAGAATCGAAAAAACACAGCAGGCA
GAAAGGTCAGTCAAAATTATCCTATAGTGCAGTCGTTCAA
CCATATCACCTAGAACCTTGAATGCATGGGTTCGTAGAGG
CAGAGGTAATACCCATGTTTACAGTCGAGAAGGCTTTTAG
CCATGTTAAATACGGTGGGGGGACATCAAGCTCGCAAATG

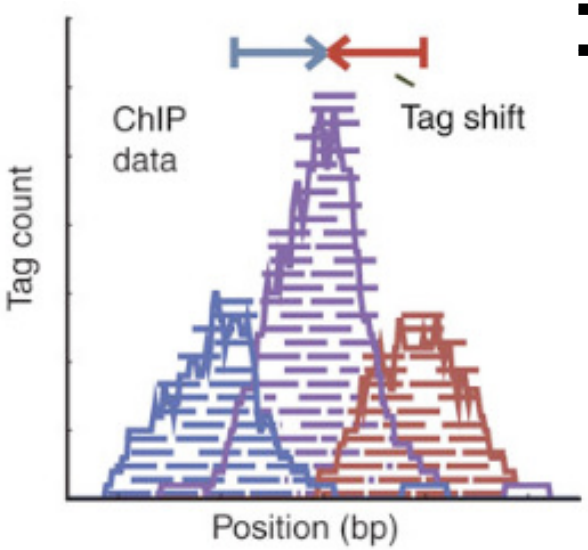
Analysis of ChIP-seq data: inferring occupancy profile along DNA



AP1 TF bound to DNA





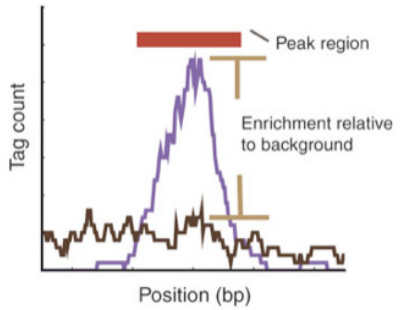


Find the distance d that maximizes the correlation function:

$$C(d) = \sum_{i=1}^{|G|} r_+(i) r_-(i+d)$$

Pepke S. et al. Nature Methods (2009)

Analysis of ChIP-seq data: finding high occupancy regions



Some notation

n_i, m_i = number of foreground/background reads in window i
N, M = total number of foreground/background reads in the sample
σ^2 = variance of multiplicative noise introduced during sample preparation
μ = depletion of background reads in bound regions
ρ = fraction of background windows
W = window size
R = range of variation of log read density in foreground vs background in a window

Probability to observe n reads in a given window – given that the window could be 'bound' or 'not bound' by the TF

$$P_b(n|m,N,M) = \frac{1}{R}$$ (uniform distribution over the possible range)

$$P_u(n|m,N,M,\mu,\sigma) = \frac{1}{\sqrt{2\pi\left(2\sigma^2 + \frac{1}{n} + \frac{1}{m}\right)}} exp\left(-\frac{\left(log\left(\frac{n}{N}\right) - log\left(\frac{m}{M}\right) - \mu\right)^2}{2\left(2\sigma^2 + \frac{1}{n} + \frac{1}{m}\right)}\right)$$

Log-likelihood of data:

$$L = \sum_i P_{mix}(n_i|m_i,N,M,\rho,\mu,\sigma) = \sum_i \rho P_u(n_i|m_i,N,M,\mu,\sigma) + (1-\rho)P_b(n_i|m_i,N,M)$$

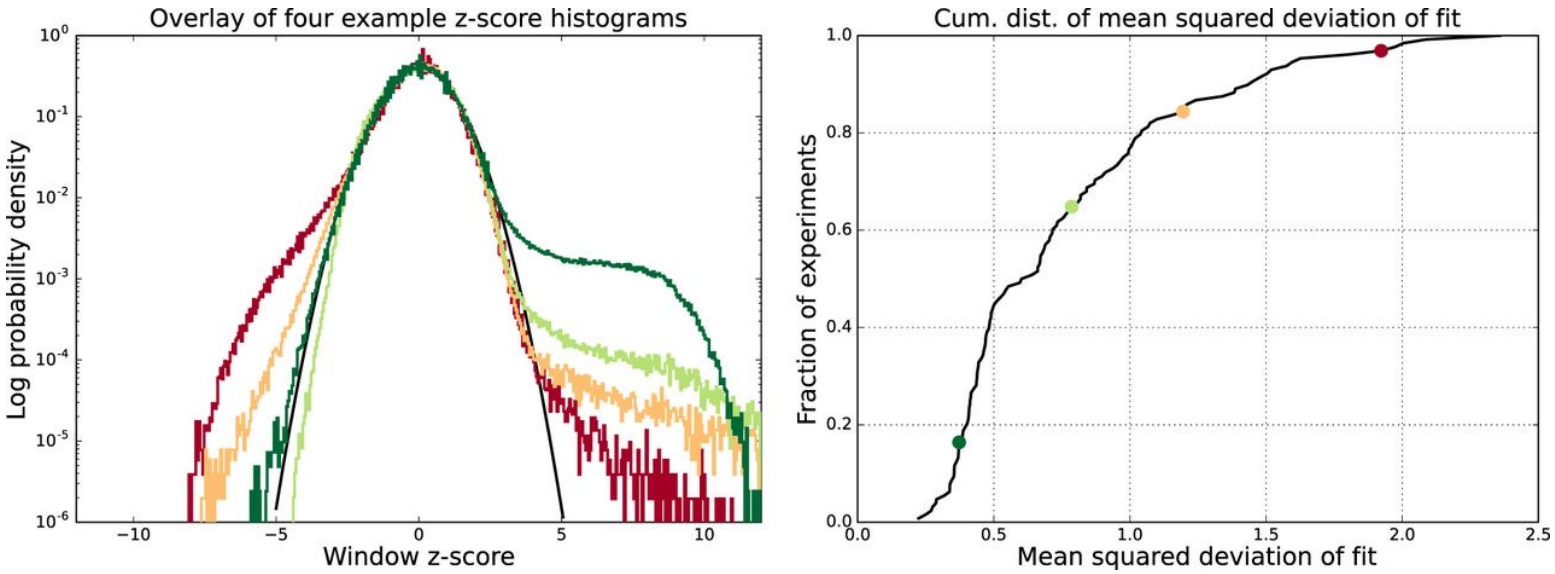
A fully automated tool:
Crunch: Integrated processing and modeling of ChIP-seq data in terms of regulatory motifs
Berger SM. Pachkov M, Arnold P, Omidi S, Kelley N, Salatino S, van Nimwegen E. *Genome Res.* 2019

Analysis of ChIP-seq data: finding high occupancy regions

After the parameters are fit, a z-score can be calculated for every window:

$$z = \frac{log\left(\frac{n}{N}\right) - log\left(\frac{m}{M}\right) - \mu}{\sqrt{2\sigma^2 + \frac{1}{n} + \frac{1}{m}}}$$

Example fits on four experimental data sets from ENCODE ChIP-seq



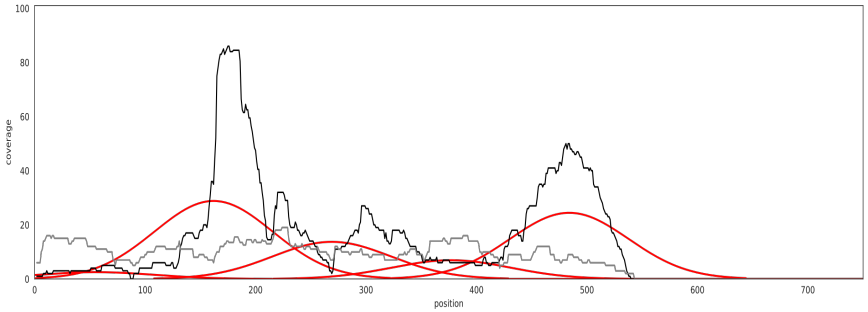
A fully automated tool:
Crunch: Integrated processing and modeling of ChIP-seq data in terms of regulatory motifs
Berger SM. Pachkov M, Arnold P, Omidi S, Kelley N, Salatino S, van Nimwegen E. *Genome Res.* 2019

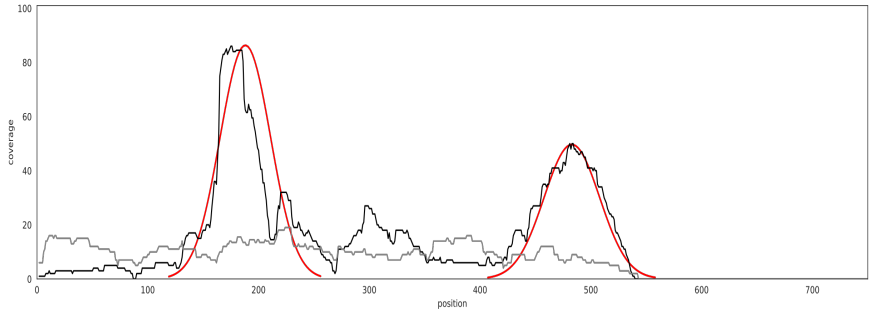
Analysis of ChIP-seq data: identifying peaks in occupancy

Modeling the coverage per position with a mixture of Gaussian peaks. Thus, for a given region

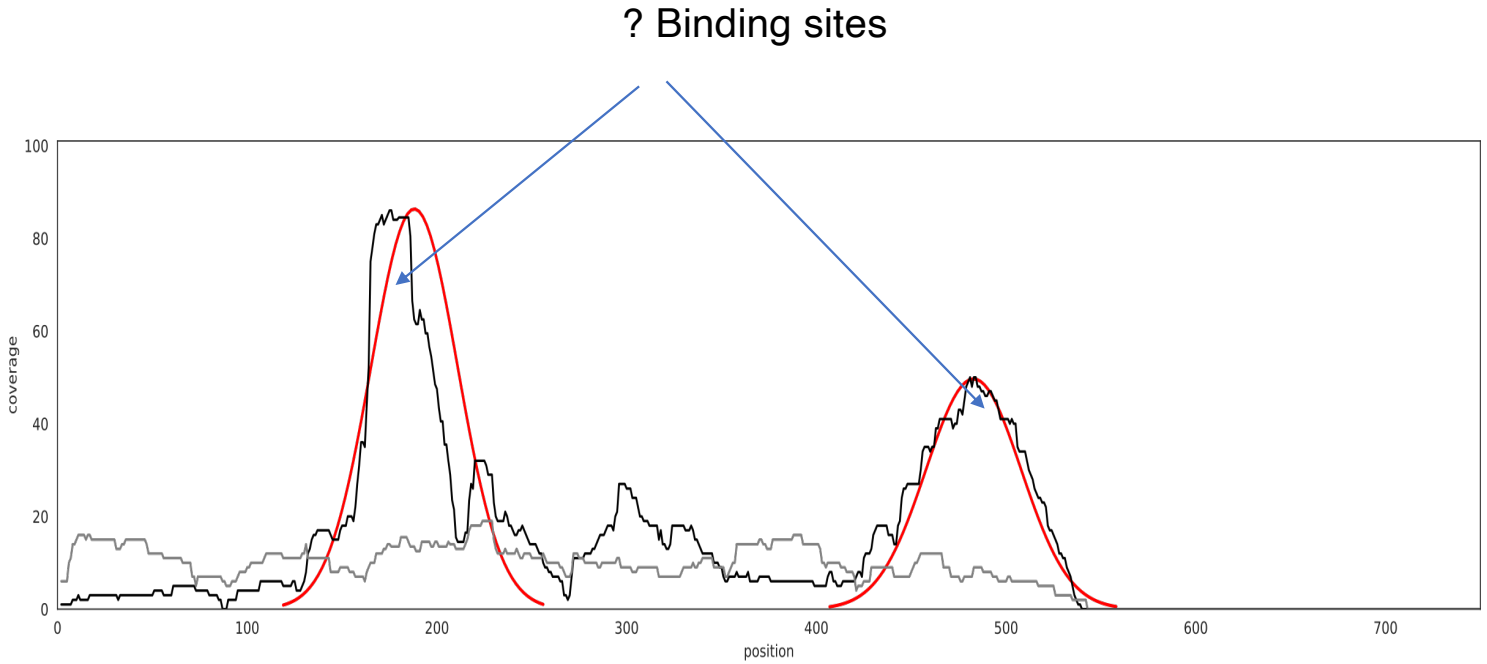
$$L(C|\vec{\pi},\vec{\sigma},\vec{\rho},W)=\prod_i\left[\rho_j\frac{1}{\sqrt{2\pi\sigma_j^2}}\exp\left(-\frac{(i-\mu_j)^2}{2\sigma_j^2}\right)+\left(1-\sum_j\rho_j\right)\frac{1}{W}\right]^{C(i)}$$

$\vec{\pi},\vec{\sigma},\vec{\rho}$ fitted by expectation maximization





Analysis of ChIP-seq data: identifying peaks in occupancy

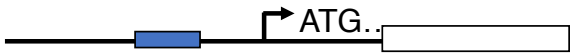


Transcription factors are generally sequence-specific

Representing binding sites

From binding energies to weight matrices

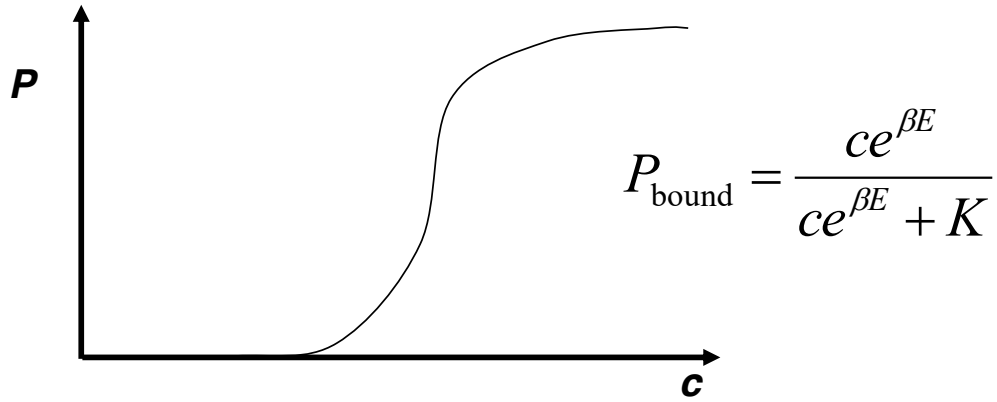
What does it mean to say that a *binding site* for a given transcription factor (TF) appears at some position in the DNA?



The interaction between the TF and the binding site is in essence characterized by two parameters:

1. The *binding energy E* of the interaction between TF and binding site
2. The concentration *c* of the transcription factor

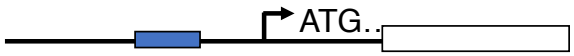
As the TF concentration increases, the fraction *P* of time the TF is bound to the site increases.



$$P_{\text{bound}} = \frac{ce^{\beta E}}{ce^{\beta E} + K}$$

From binding energies to weight matrices

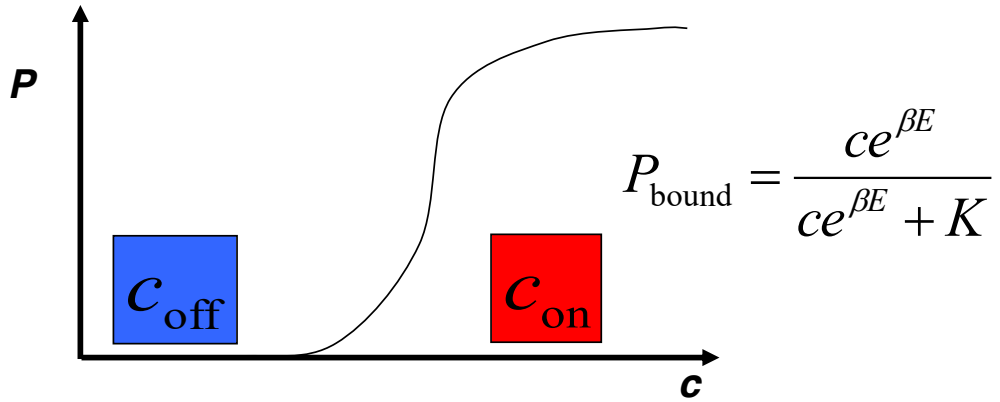
What does it mean to say that a *binding site* for a given transcription factor (TF) appears at some position in the DNA?



The interaction between the TF and the binding site is in essence characterized by two parameters:

1. The *binding energy E* of the interaction between TF and binding site
2. The concentration *c* of the transcription factor

As the TF concentration increases, the fraction *P* of time the TF is bound to the site increases.

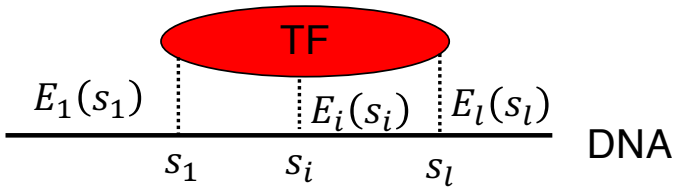


If the TF's concentration goes between a low 'off' state and a high 'on' state, then binding sites are characterized by binding energies *E* such that they will be mostly unbound in the 'off' state and mostly bound in the 'on' state.

From binding energies to weight matrices

1. We assume the binding energy of a sequence *s* is an additive function of the individual bases:

$$E(s) = \sum_{i=1}^{l} E_i(s_i)$$



2. The probability for the site to be bound can be roughly described by the following function of energy *E*(*s*) and concentration of the transcription factor *c*

$$P_{bound}(s) = \frac{ce^{\beta E(s)}}{ce^{\beta E(s)} + K}$$

3. Assume that the only constraint on 'functional binding sites' is that they have some characteristic *average energy* E. Then we get using maximum entropy:

$$P(s) = \frac{e^{\lambda E(s)}}{\sum_{s'} e^{\lambda E(s')}} = \prod_{i=1}^{l} \frac{e^{\lambda E_i(s_i)}}{\sum_{\alpha} e^{\lambda E_i(\alpha)}}$$

where the Lagrange multiplier λ is chosen such that $\sum_s E(s)P(s) = E$

From binding energies to weight matrices

$$P(s) = \frac{e^{\lambda E(s)}}{\sum_{s'} e^{\lambda E(s')}} = \prod_{i=1}^{l} \frac{e^{\lambda E_i(s_i)}}{\sum_{\alpha} e^{\lambda E_i(\alpha)}}$$

This can be rewritten in terms of a weight matrix (WM) w.

$$P(s) = \prod_{i=1}^{l} P_i(s_i) \stackrel{\text{def}}{=} \prod_{i=1}^{l} w_{s_i}^i \qquad\qquad w_\alpha^i = \frac{e^{\lambda E_i(\alpha)}}{\sum_{\alpha'} e^{\lambda E_i(\alpha')}}$$

The probability that a binding site for the TF will have a sequence s is given by:

$$P(s|w) = \prod_{i=1}^{l} w_{s_i}^i$$

Note: This is **not** the probability that a segment with sequence s is a binding site!

Promoter regions recognized by σ70 subunit of E.coli RNA polymerase

http://cwx.prenhall.com/horton/medialib/media_portfolio/

Sequence specificity of σ_{70}

$$w_b^i = \frac{n_b^i + \alpha_b^i}{\sum_b \left(n_b^i + \alpha_b^i \right)}$$

n_b^i count of nucleotide *b* at position *i*

α_b^i pseudo-count of nucleotide *b* at position *i* (here 0.25)

Sites\Position	1	2	3	4	5	6
1	T	T	G	A	C	T
2	T	T	G	A	C	A
3	T	T	G	A	C	A
4	T	T	T	A	C	A
5	T	T	T	A	C	A
6	C	T	G	A	C	G
7	T	T	G	T	C	A
8	T	T	G	A	C	T
9	C	A	C	A	C	T
10	A	T	G	T	C	A

Nucl.\Pos.	1	2	3	4	5	6
A	1	1	0	8	0	6
C	2	0	1	0	10	0
G	0	0	7	0	0	1
T	7	9	2	2	0	3

Nucl.\Pos.	1	2	3	4	5	6
A	0.11	0.11	0.02	0.75	0.02	0.57
C	0.20	0.02	0.11	0.02	0.93	0.02
G	0.02	0.02	0.66	0.02	0.02	0.11
T	0.66	0.84	0.20	0.20	0.02	0.30

Sequence specificity of σ_{70}

Nucl.\Pos.	1	2	3	4	5	6
A	1	1	0	8	0	6
C	2	0	1	0	10	0
G	0	0	7	0	0	1
T	7	9	2	2	0	3

$$w_b^i = \frac{n_b^i + \alpha_b^i}{\sum_b \left(n_b^i + \alpha_b^i \right)}$$

n_b^i count of nucleotide *b* at position *i*

α_b^i pseudo-count of nucleotide *b* at position *i* (here 0.25)

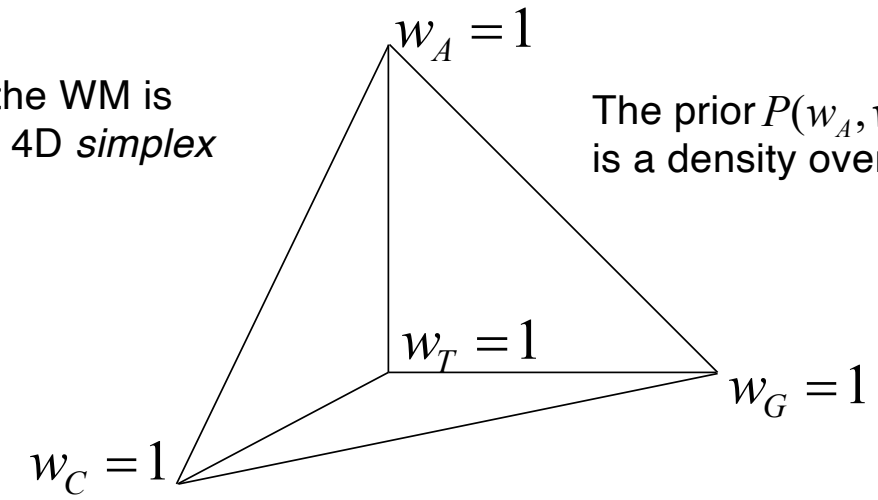
How do we set the pseudo-count?

$$P(S \mid w) = \prod_{s \in S} P(s \mid w) = \prod_{s \in S} \left[\prod_{i=1}^{l} w_{s_i}^i \right] = \prod_{i=1}^{l} \left[\prod_{\alpha} \left(w_\alpha^i \right)^{n_\alpha^i} \right]$$

To calculate $P(w|S)$ we need a prior $P(w)$.

Prior over WMs

A column of the WM is
a point in the 4D *simplex*



The prior $P(w_A, w_C, w_G, w_T)$
is a density over the simplex

The family of *Dirichlet* priors: $P(w)dw = \frac{\Gamma(4\gamma)}{[\Gamma(\gamma)]^4} \prod_\alpha (w_\alpha)^{\gamma-1} dw$

- The case $\gamma = 1$ corresponds to the uniform prior.
- For $\gamma < 1$ more weight is on the corners and edges of the simplex, i.e. one expects a distribution heavily biased to one or two bases.
- For $\gamma > 1$ more weight is on the middle of the simplex, i.e. one expects all bases to have equal probabilities.

Inferring a WM from a set of sites

$$P(w)dw = \frac{\Gamma(4\gamma)}{[\Gamma(\gamma)]^4}\prod_{\alpha}(w_\alpha)^{\gamma-1}dw \qquad P(S,w) = P(S \mid w)P(w) = \prod_{i=1}^{I}\left[\Gamma(4\gamma)\prod_{\alpha}\frac{(w_\alpha^i)^{n_\alpha^i+\gamma-1}}{\Gamma(\gamma)}\right]$$

Posterior: $P(w \mid S) = \dfrac{P(S \mid w)P(w)}{P(S)}$

The normalization constant is: $P(S) = \int P(S \mid w)P(w)\,dw$

This is the probability all sequences in *S* come from one WM, irrespective of what that WM is.

Weight matrix representation

Information score of a weight matrix is a measure of the specificity of the binding factor.

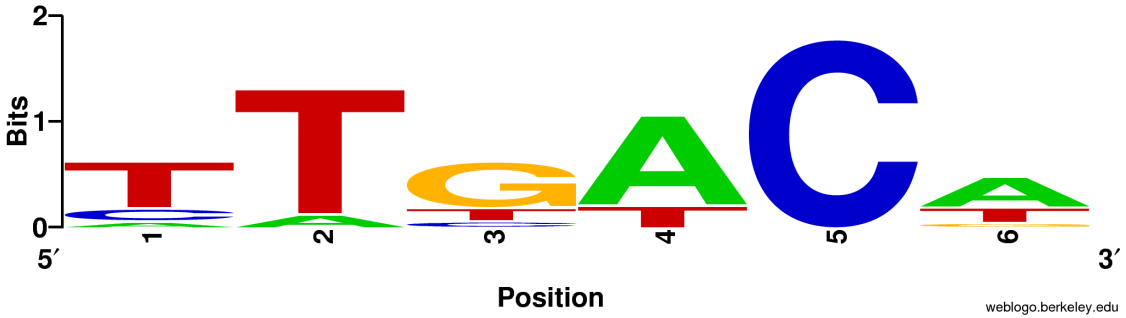
$$I = \sum_{i,b} f_b^i \; log\left(\frac{f_b^i}{p_b}\right)$$

$f_b^i = \frac{n_b^i}{\sum_b n_b^i}$ and p_b is the background frequency of nucleotide $\boldsymbol{i}$.

I = 0, if $f_b^i = p_b$ for every b and i.

I = maximal if $f_b^i = 1$ for some b, and $f_b^i = 0$ for all others.

Nucl.\Pos.	1	2	3	4	5	6
A	0.11	0.11	0.02	0.75	0.02	0.57
C	0.20	0.02	0.11	0.02	0.93	0.02
G	0.02	0.02	0.66	0.02	0.02	0.11
T	0.66	0.84	0.20	0.20	0.02	0.30



$$I = 0.11\; log\left(\frac{0.11}{0.25}\right) + 0.2\; log\left(\frac{0.2}{0.25}\right) +$$
$$0.02\; log\left(\frac{0.02}{0.25}\right) + 0.66\; log\left(\frac{0.66}{0.25}\right) + \cdots = 3.87$$

In bits: 5.58

How do we use weight matrices to find binding sites?

Computing the likelihood of a sequence given a weight matrix

Nucl.\Pos.	1	2	3	4	5	6
A	0.11	0.11	0.02	0.75	0.02	0.57
C	0.20	0.02	0.11	0.02	0.93	0.02
G	0.02	0.02	0.66	0.02	0.02	0.11
T	0.66	0.84	0.20	0.20	0.02	0.30

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT

$$P(s[i..i+\omega-1]\,|\,\vec{w}) = \prod_{j=1}^{\omega} w^{j}_{s[i+j]}$$

ω is the length of ***w***

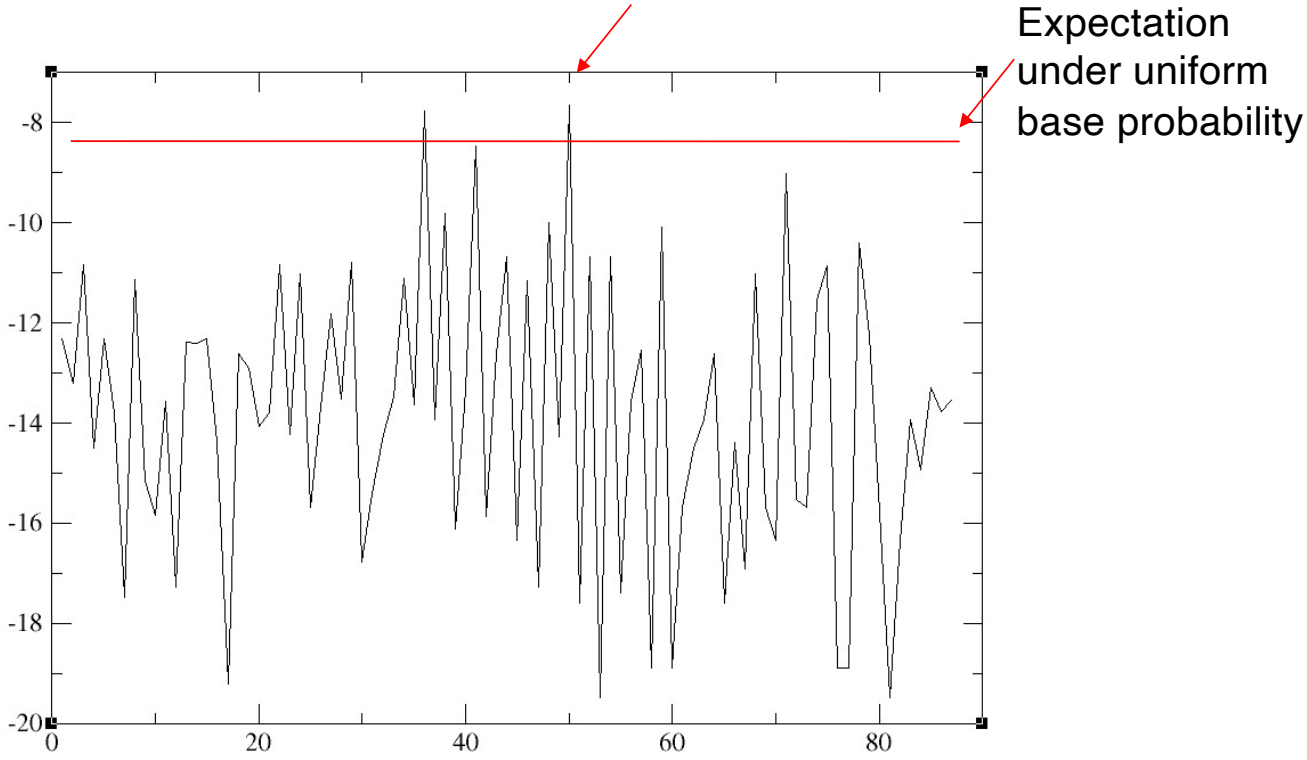
$$P(\text{TGCGCA}|\vec{w}) = e^{-11.08}$$

$$P(\text{TGCGCA}|\text{"chance"}) = 0.25^{6} = e^{-8.32}$$

How do we use weight matrices to find binding sites?

Log-likelihood profile of the sequence

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT



Multiple sites and posterior probabilities of sites

Assume that we have weight matrices for two transcription factors, TF1 and TF2, and that we have some model for the background sequence, B.

Consider one way of placing weight matrices and background regions over a sequence (this is called a ***parse***):

$P(s[3..17|\overrightarrow{w_2})$ $P(s[29..38|\overrightarrow{w_1})$ $P(s[61..70|\overrightarrow{w_1})$ $P(s[71..71|\overrightarrow{w_B})$

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT

We can calculate the likelihood of this particular parse using, for each region,

$$P(s[i..i+\omega-1|\vec{w}) = \prod_{j=1}^{\omega} w^{j}_{s_{i+j}}$$

where $\vec{w}$ is either the weight matrix corresponding to the transcription factor bound at i or to the background, and ω is the length of the weight matrix.

Multiple sites and posterior probabilities of sites

Total likelihood of the sequence, summed over all the possible parses, represents the partition function (usually denoted by **Z**). It can be calculated using a dynamic programming technique.

$$Z(1,i) = \sum_{j=1}^{m} \left(Z\left(1, i-\omega_j\right) P(s[i-\omega_j+1..i|\vec{w}^j) \right)$$

This is a summation over all ***m*** weight matrices of the contribution of parses ending with a given weight matrix.

Posterior probability to have a site for TF_i starting at position i:

$$P(TF_2 \; at \; j) = \frac{Z(1,i-1)P(s[i..i+\omega_2-1|\vec{w}^2)Z(i+\omega_2,L)}{Z(1,L)}$$

Multiple sites and posterior probabilities of sites

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT

i

$$Z(1,i) = \sum_{j=1}^{m} \left(Z\left(1, i-\omega_j\right) P(s[i-\omega_j+1..i|\vec{w}^j) \right)$$

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGATGATGGATGGCCCGGTACCGTT

i

$$P(TF_2 \; at \; j) = \frac{Z(1,i-1)P(s[i..i+\omega_2-1|\vec{w}^2)Z(i+\omega_2,L)}{Z(1,L)}$$

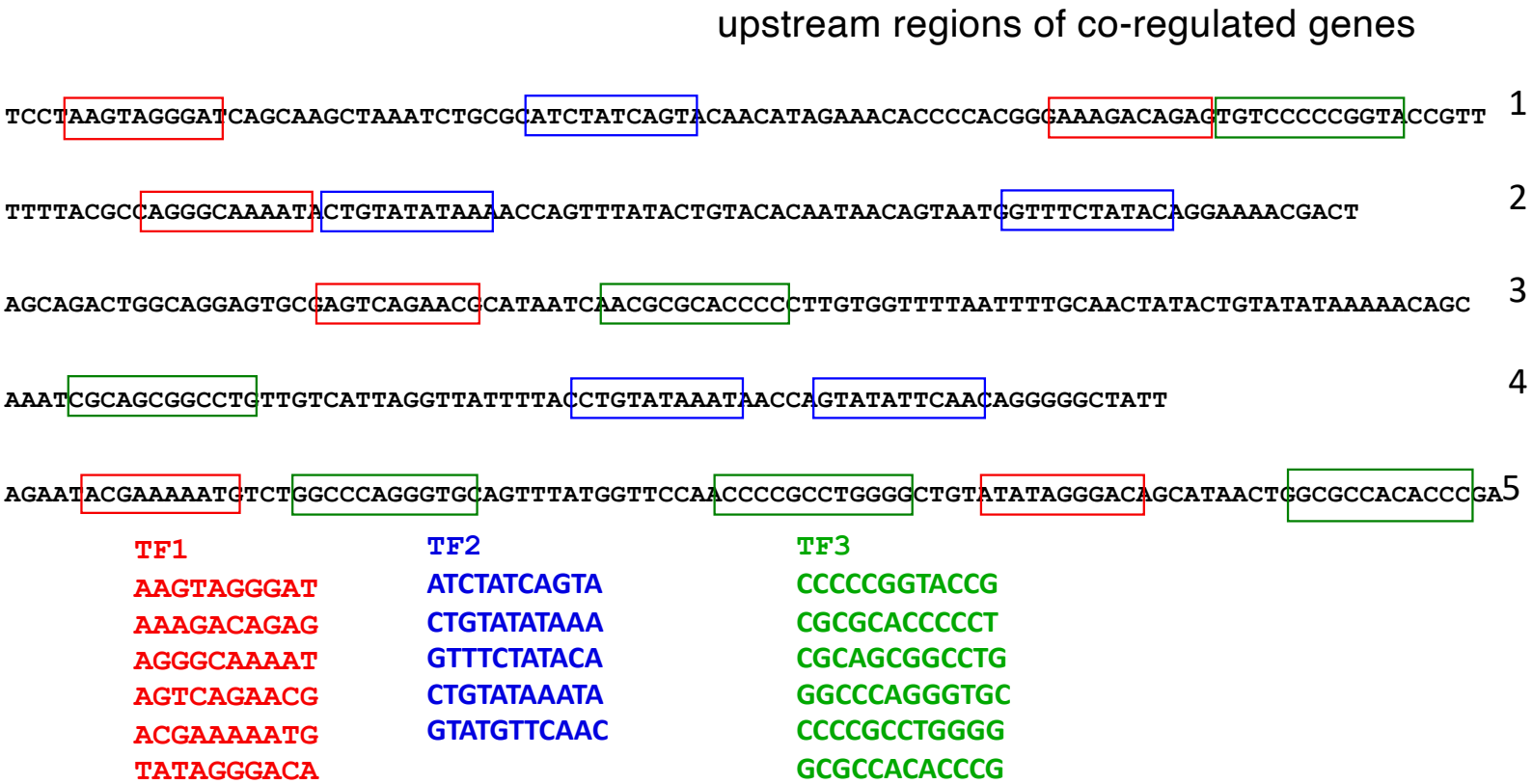
Inferring novel binding specificities

Inferring binding specificities from data about co-regulated genes

Example setup: we identified regions that are bound by a given transcription factor whose specificity we don't know.

Question: find the binding specificity of the transcription factor, represented as a weight matrix, from this set of binding sites.

Inferring binding specificities from data about co-regulated genes



Inferring binding specificities from data about co-regulated genes

Focus on a single transcription factor

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGACAGAGTGTCCCCCGGTACCGTT 1

TTTTACGCCAGGGCAAAATACTGTATATAAAACCAGTTTATACTGTACACAATAACAGTAATGGTTTCTATACAGGAAAACGACT 2

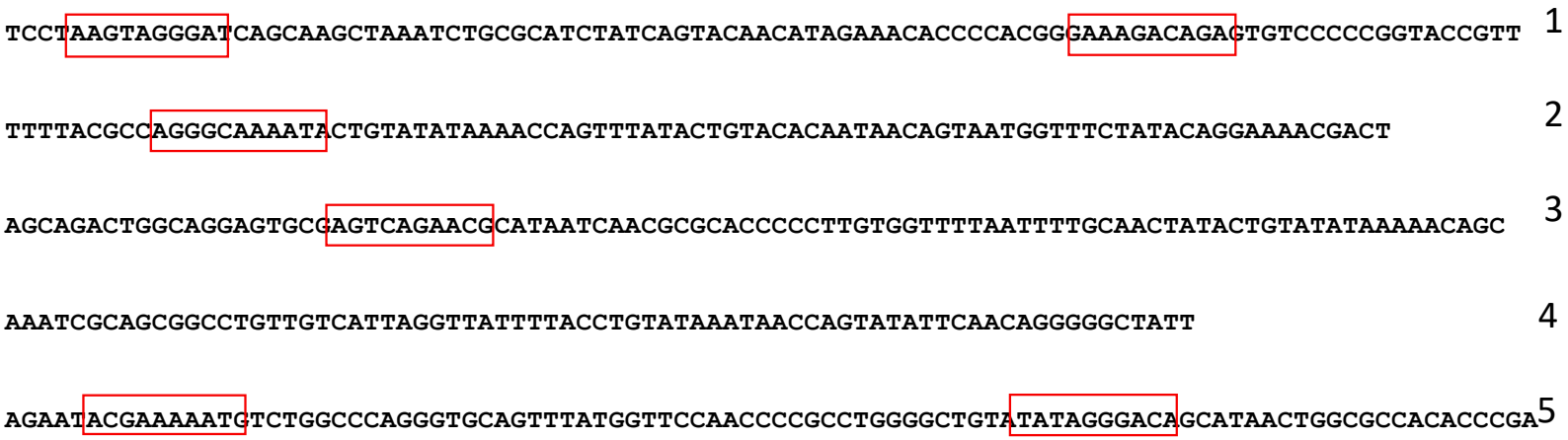
AGCAGACTGGCAGGAGTGCGAGTCAGAACGCATAATCAACGCGCACCCCCTTGTGGTTTTAATTTTGCAACTATACTGTATATAAAAACAGC 3

AAATCGCAGCGGCCTGTTGTCATTAGGTTATTTTACCTGTATAAATAACCAGTATATTCAACAGGGGGCTATT 4

AGAATACGAAAAATGTCTGGCCCAGGGTGCAGTTTATGGTTCCAACCCCGCCTGGGGCTGTATATAGGGACAGCATAACTGGCGCCACACCCGA 5

Inferring binding specificities from data about co-regulated genes

Focus on a single transcription factor
What we would like to find:



Inferring binding specificities from data about co-regulated genes

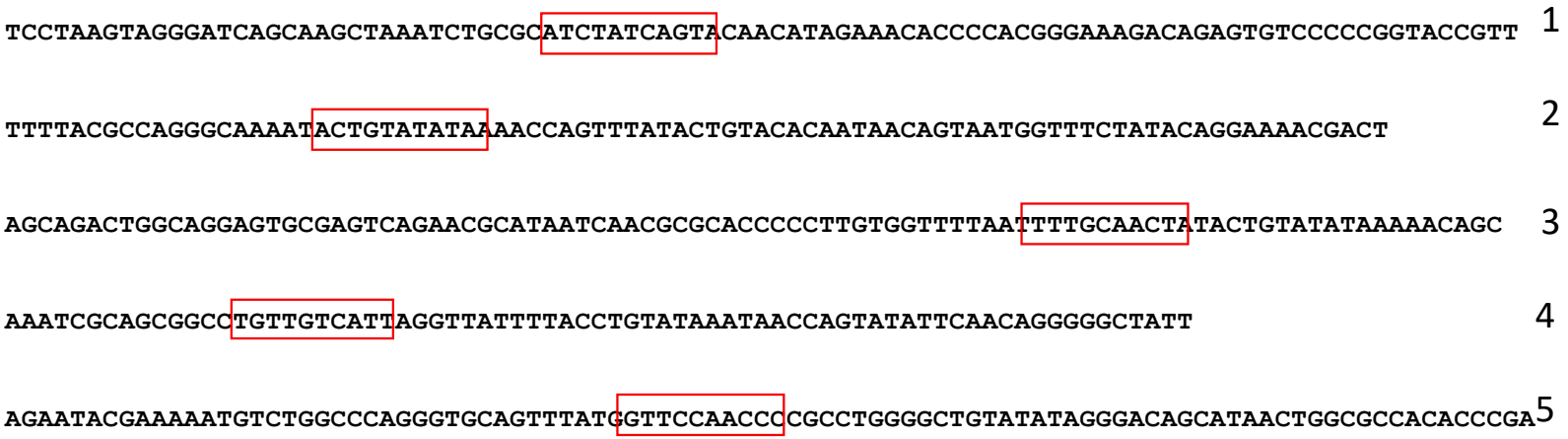
Gibbs sampling algorithm for inferring binding specificity

Start with windows placed randomly, one in each sequence

Inferring binding specificities from data about co-regulated genes

Gibbs sampling algorithm for inferring binding specificity

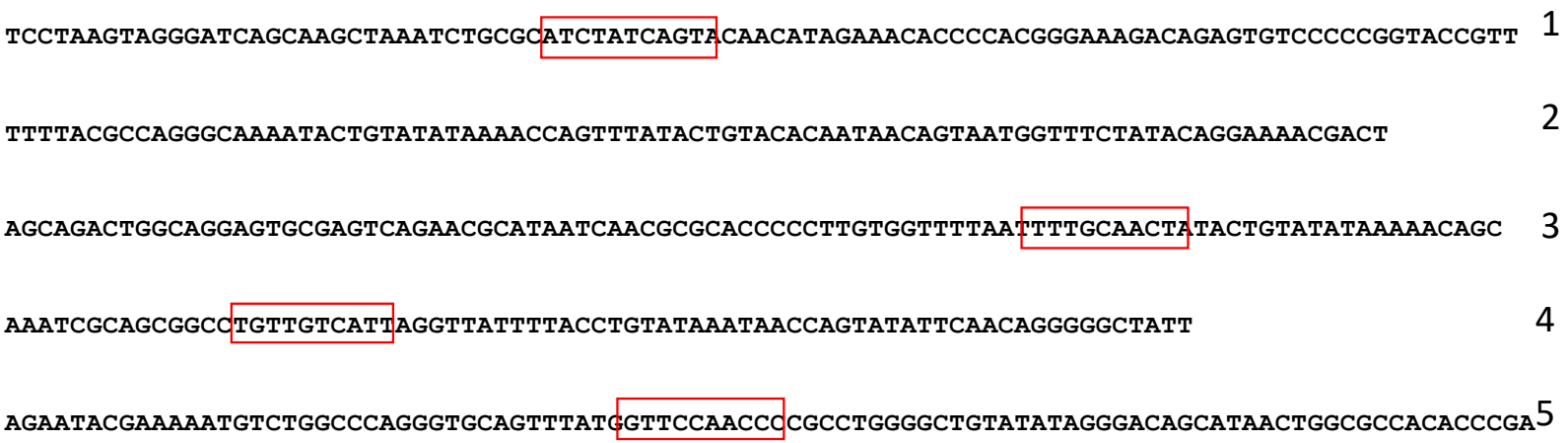
Start with windows placed randomly, one in each sequence



Inferring binding specificities from data about co-regulated genes

At each iteration:

remove one window

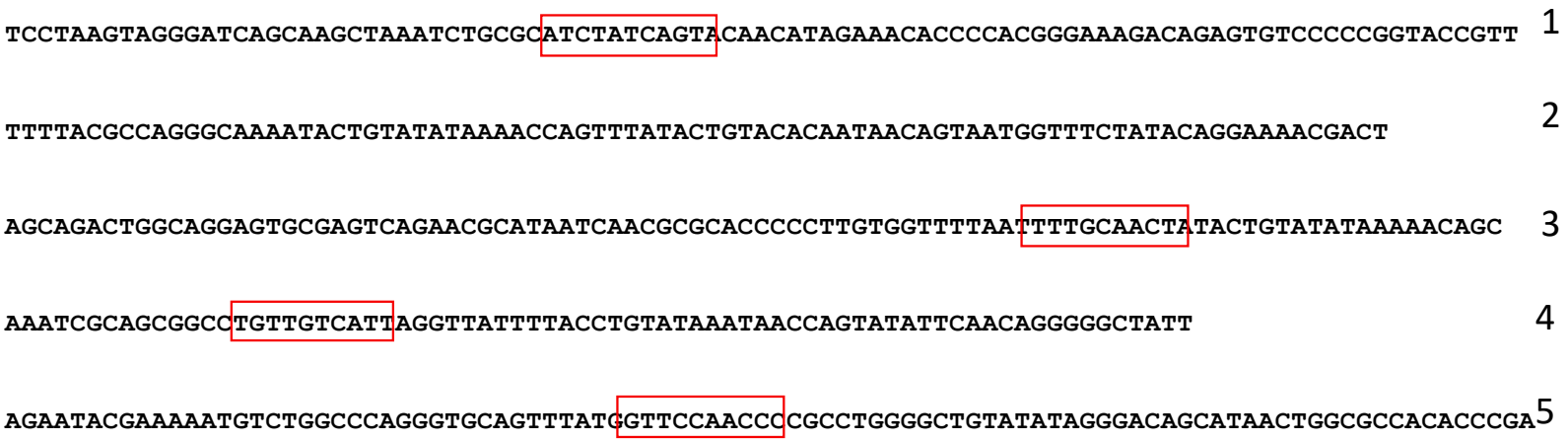


Inferring binding specificities from data about co-regulated genes

At each iteration:

- remove one window
- use the others to infer WM

```
ATCTATCAGT
TTTGCAACTA
TGTTGTCATT
GTTCCAACCC
```



Inferring binding specificities from data about co-regulated genes

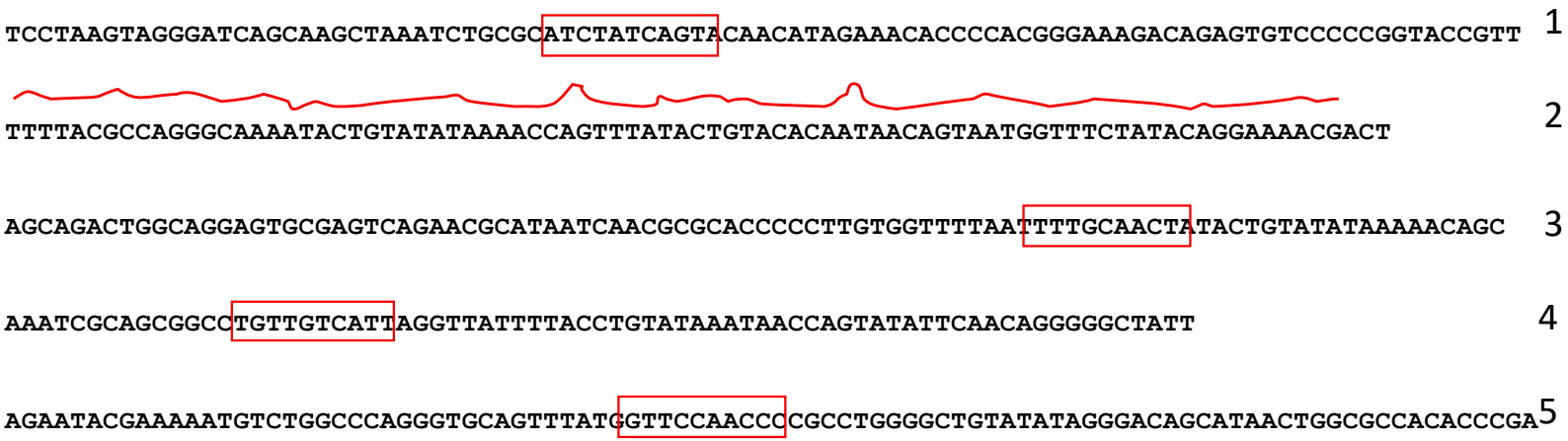
At each iteration:

remove one window

use the others to infer WM

```
ATCTATCAGT
TTTGCAACTA
TGTTGTCATT
GTTCCAACCC
```

at each position in sequence 2 calculate the probability that the subsequence starting at that position was generated from the weight matrix



Inferring binding specificities from data about co-regulated genes

At each iteration:

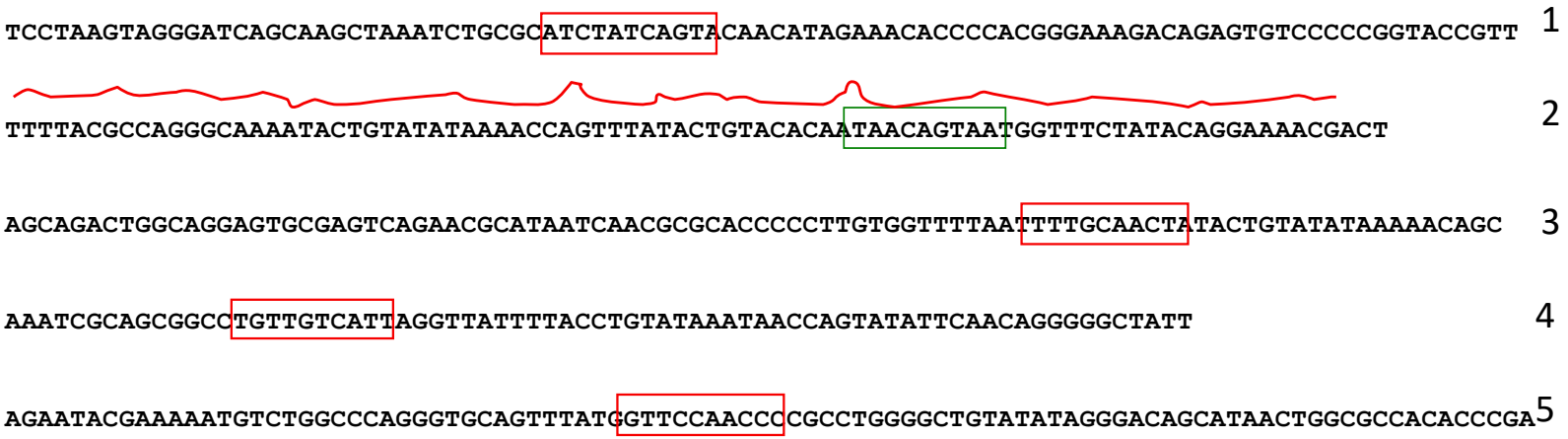
remove one window

use the others to infer WM

```
ATCTATCAGT
TTTGCAACTA
TGTTGTCATT
GTTCCAACCC
```

at each position in sequence 2 calculate the probability that the subsequence starting at that position was generated from the weight matrix

sample one window according to these probabilities



Sampling probability distributions using the Metropolis algorithm

Each way of placing a window of length ***L*** in each of the n upstream regions represents a state **x**, with an associated probability, **P(x).**

P(x) is proportional to the likelihood that all subsequences enclosed by the windows are generated using the same weight matrix.

To find states with high probability, we sample **P(x)** using a Markov chain:

1. given state **x**, state **y** is proposed with probability **P(y|x)**, such that **P(y|x)** = **P(x|y)**.
2. state **y** is accepted with probability min(1, **P(y)**/**P(x)**).

If these constraints are met, it can be shown that each state is visited in proportion to its probability.

Inferring binding specificities from data about co-regulated genes

MEME algorithm for inferring sequence specificity

Start with a guess for a weight matrix and for the probability of occurrence of binding sites

TCCTAAGTAGGGATCAGCAAGCTAAATCTGCGCATCTATCAGTACAACATAGAAACACCCCACGGGAAAGACAGAGTGTCCCCCGGTACCGTT 1

TTTTACGCCAGGGCAAAATACTGTATATAAAACCAGTTTATACTGTACACAATAACAGTAATGGTTTCTATACAGGAAAACGACT 2

AGCAGACTGGCAGGAGTGCGAGTCAGAACGCATAATCAACGCGCACCCCCTTGTGGTTTTAATTTTGCAACTATACTGTATATAAAAACAGC 3

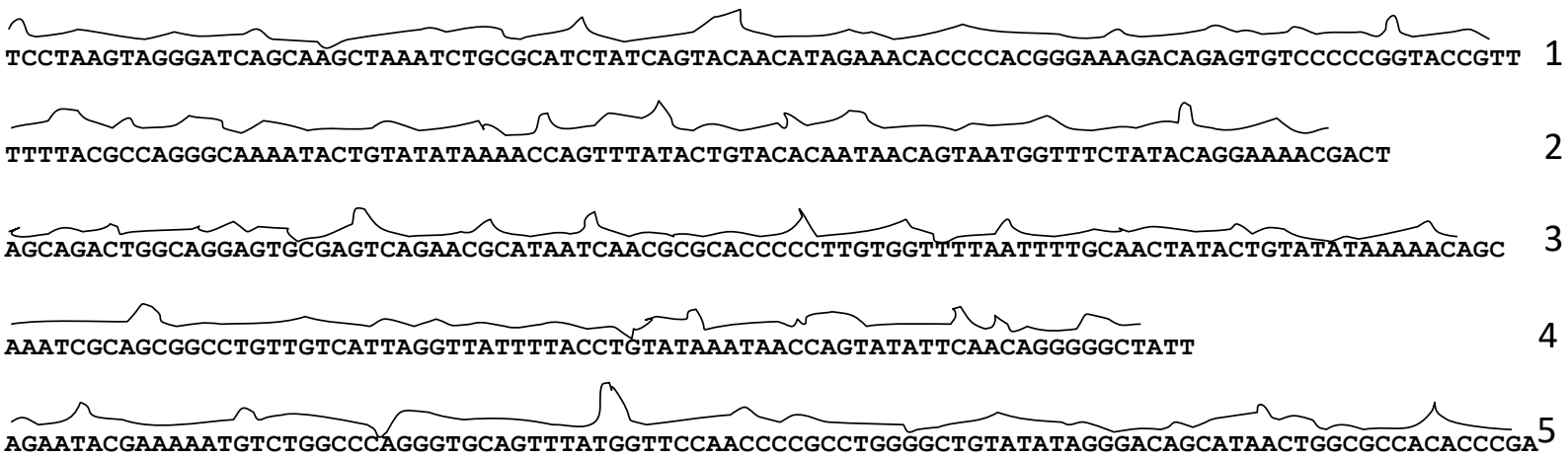
AAATCGCAGCGGCCTGTTGTCATTAGGTTATTTTACCTGTATAAATAACCAGTATATTCAACAGGGGGCTATT 4

AGAATACGAAAAATGTCTGGCCCAGGGTGCAGTTTATGGTTCCAACCCCGCCTGGGGCTGTATATAGGGACAGCATAACTGGCGCCACACCCGA 5

Inferring binding specificities from data about co-regulated genes

At each iteration

use current weight matrix and prior probability for binding sites to calculate the probability at each position in every sequence that there is a binding site starting at that position

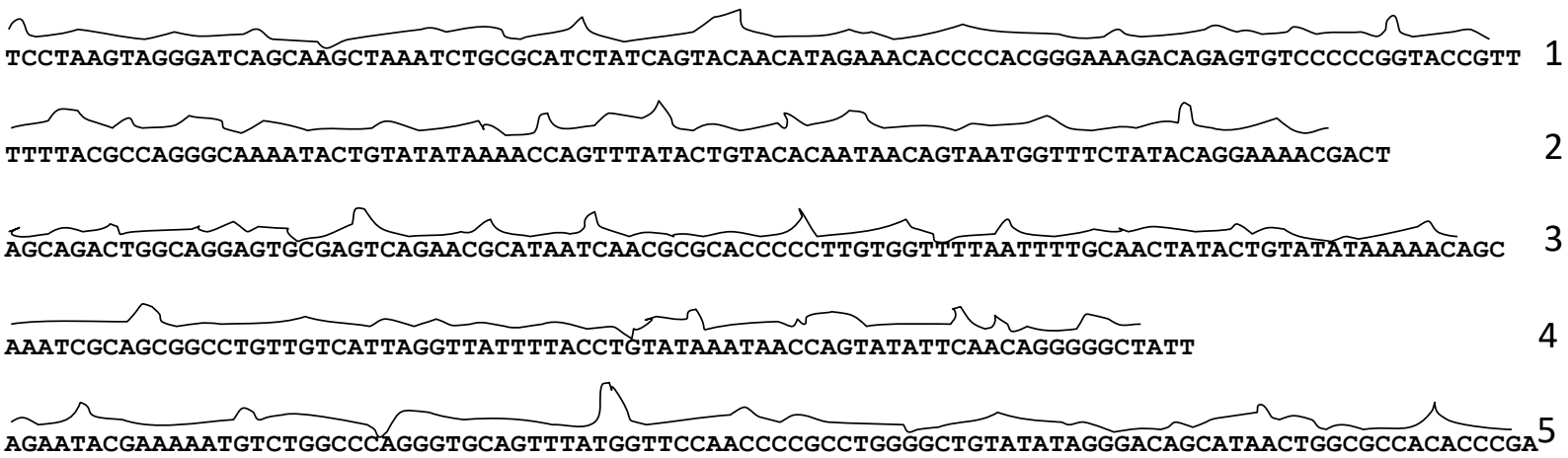


Inferring binding specificities from data about co-regulated genes

At each iteration

use current weight matrix and prior probability for binding sites to calculate the probability at each position in every sequence that there is a binding site starting at that position

update weight matrix and prior probability of sites



Inferring binding specificities from data about co-regulated genes

Posterior probability of a site starting at position j :

$$P(site\ at\ j) = \frac{\pi_t P(s[j..j+\omega-1|\vec{w})}{\pi_t P(s[j..j+\omega-1|\vec{w}) + (1-\pi_t)P(s[j..j+\omega-1|\vec{B})}$$

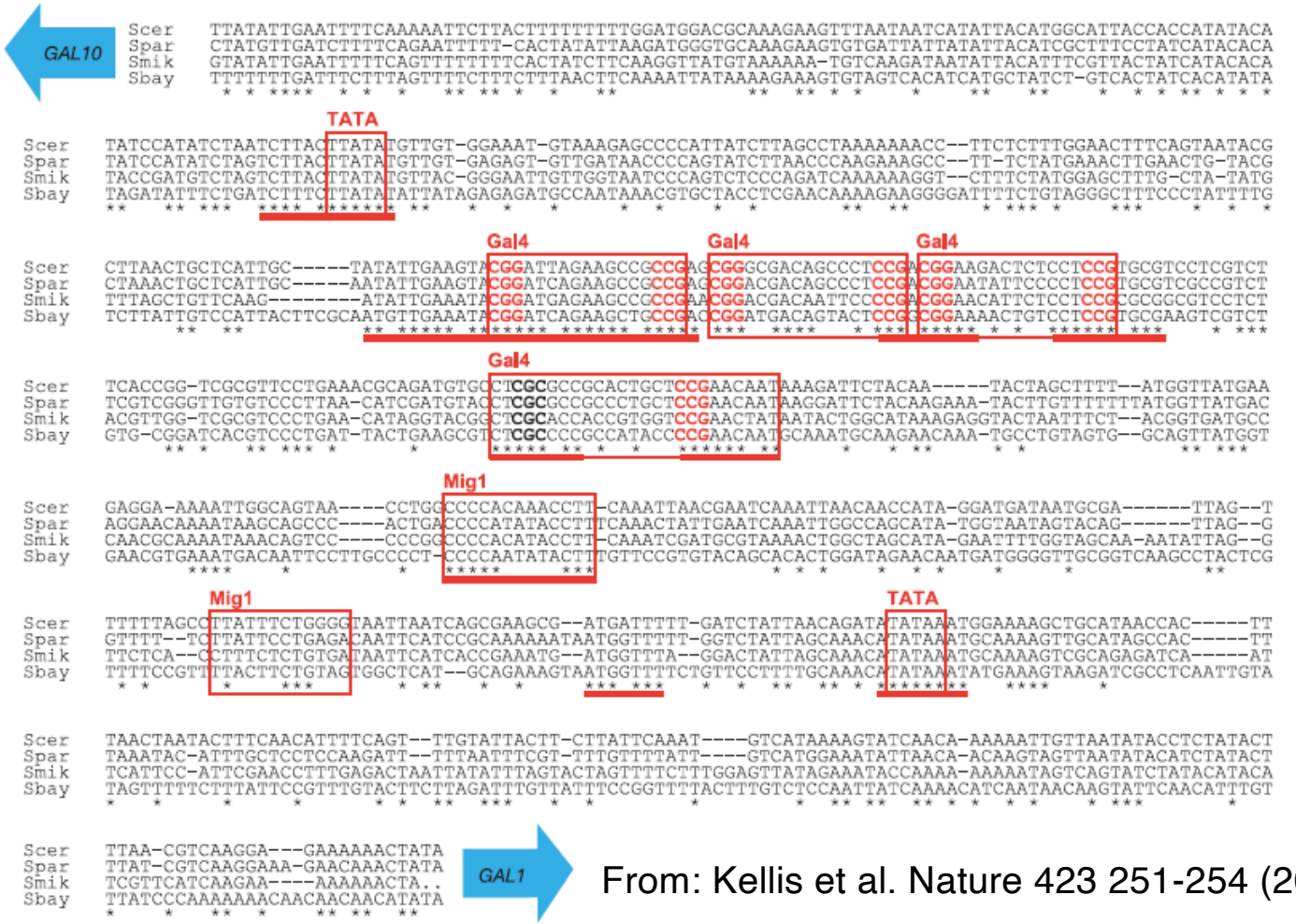
Updating prior probability of binding sites:

$$\pi_{t+1} = \frac{\sum_j P(site\ at\ j)}{L}$$

Updating the weight matrix entries:

$$w_\alpha^k\Big|_{t+1} = \frac{\sum_j P(site\ at\ j)\delta(s[j+k],\alpha)}{\sum_j P(site\ at\ j)}$$

Phylogenetic Footprinting



From: Kellis et al. Nature 423 251-254 (2003)

Ab initio discovery of regulatory sites

Approaches we have discussed:

1. Collect sets of (intergenic) sequences that are thought to contain binding sites for a common regulatory factor. Examples:
   - Upstream regions of co-regulated genes
   - Sequence fragments pulled down with ChIP.

   then search for overrepresented short sequence motifs.

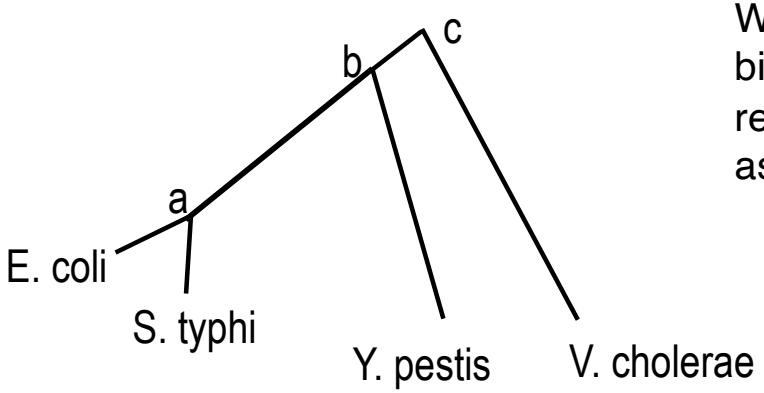
2. Phylogenetic footprinting: create multiple alignments of orthologous intergenic sequences and identify sequence segments more conserved than “average”.

Can we combine these two procedures into one rigorous approach?

Sampling alignments of related sequences: the Phylogibbs algorithm

```
E. Coli      acgtaactagtga
S. Typhi     acgttgctagatg
Y. Pest      tcgttgctataat
V. Cholerae  aggtagcgagaag
```

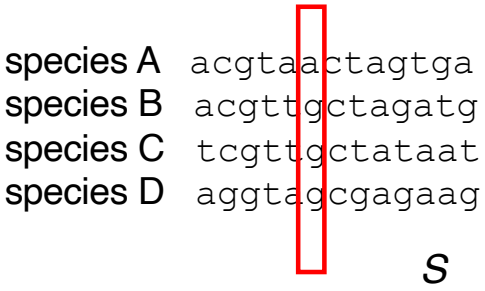
The nucleotides in one column are not independent samples of the weight matrix, but are phylogenetically related.

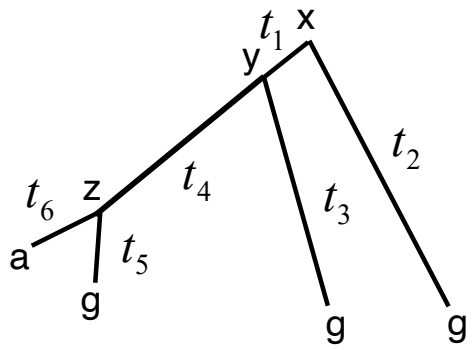


When accounting for the bases contained in binding sites, the sites that are phylogenetically related should not be counted independently but as an “effective” number of sites.

PhyloGibbs: A Gibbs Sampling Motif Finder That Incorporates Phylogeny
Siddharthan R, Siggia ED, van Nimwegen E
PLoS Computational Biology Vol. 1, No. 7, e67 doi:10.1371/journal.pcbi.0010067

Probability of an alignment column





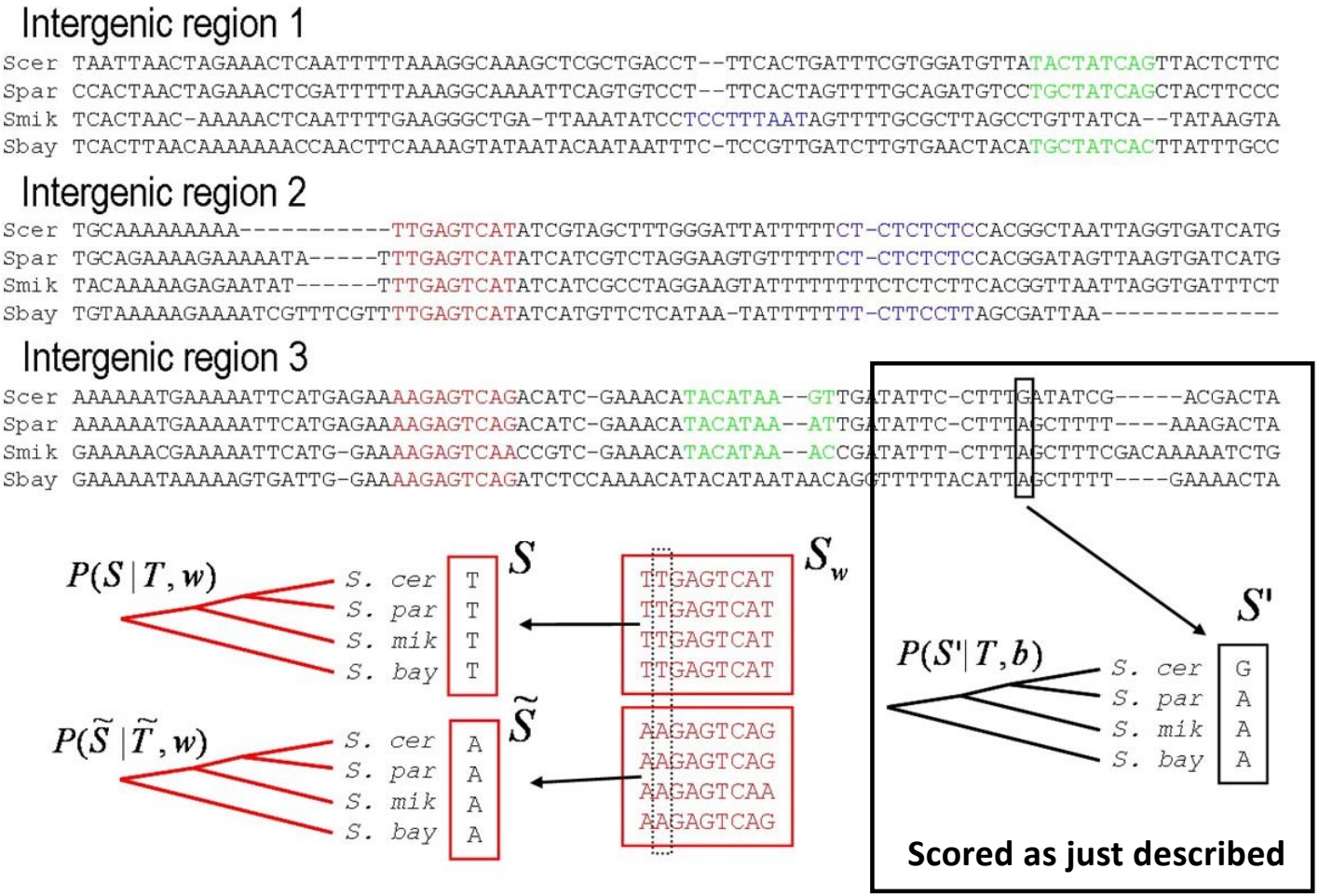
Phylogenetic tree relating the species

The probability *P(S|T, w)* of the bases at the leaves given the tree *T* and the limit frequencies *w* is the product over the transition probabilities along each of the branches, summed over the possible bases at the internal nodes:

$$P(S \mid T, w) = \sum_{x,y,z} w_x P(y \mid x, t_1) P(g \mid x, t_2) P(g \mid y, t_3) P(z \mid y, t_4) P(g \mid z, t_5) P(a \mid z, t_6)$$

Recall: we can use the recursion relation (Felsenstein, 1981) introduced earlier.

PhyloGibbs: generalized motif sampling on phylogenetically related sequences



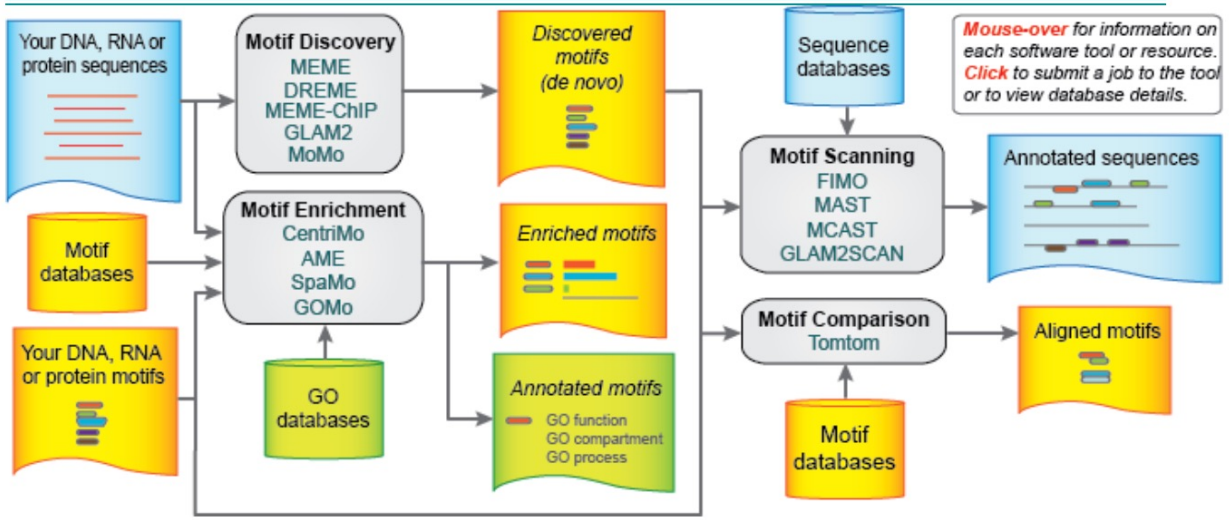
We have to deal with the WM being unknown! Integrate over WMs

meme-suite.org

The MEME Suite

Motif-based sequence analysis tools

- MEME Suite 5.1.1
- ▶Motif Discovery
- ▶Motif Enrichment
- ▶Motif Scanning
- ▶Motif Comparison
- ▶Gene Regulation
- ▶Manual
- ▶Guides & Tutorials
- ▶Sample Outputs
- ▶File Format Reference
- ▶Databases
- ▶Download & Install
- ▶Help
- ▶Alternate Servers
- ▶Authors & Citing
- ▶Recent Jobs
- ↪ Previous version 5.1.0



MEME — Improved
Multiple Em for Motif Elicitation

CentriMo
Local Motif Enrichment Analysis

FIMO
Find Individual Motif Occurrences

DREME
Discriminative Regular Expression Motif Elicitation

AME — Improved
Analysis of Motif Enrichment

MAST
Motif Alignment & Search Tool

MEME-ChIP — Improved
Motif Analysis of Large Nucleotide Datasets

SpaMo
Spaced Motif Analysis Tool

MCAST
Motif Cluster Alignment and Search Tool

GLAM2
Gapped Local Alignment of Motifs

GOMo
Gene Ontology for Motifs

GLAM2Scan
Scanning with Gapped Motifs

MoMo — Improved
Modification Motifs

Tomtom
Motif Comparison Tool

GT-Scan
Identifying Unique Genomic Targets

T-Gene — New
Predicting Target Genes

Phylogibbs Online

www.phylogibbs.unibas.ch

PhyloGibbs is an algorithm for discovering regulatory sites in a collection of DNA sequences, including multiple alignments of orthologous sequences from related organisms. Many existing approaches to either search for sequence-motifs that are overrepresented in the input data, or for sequence-segments that are more conserved evolutionary than expected. PhyloGibbs combines these two approaches and identifies significant sequence-motifs by taking both over-representation and conservation signals into account.

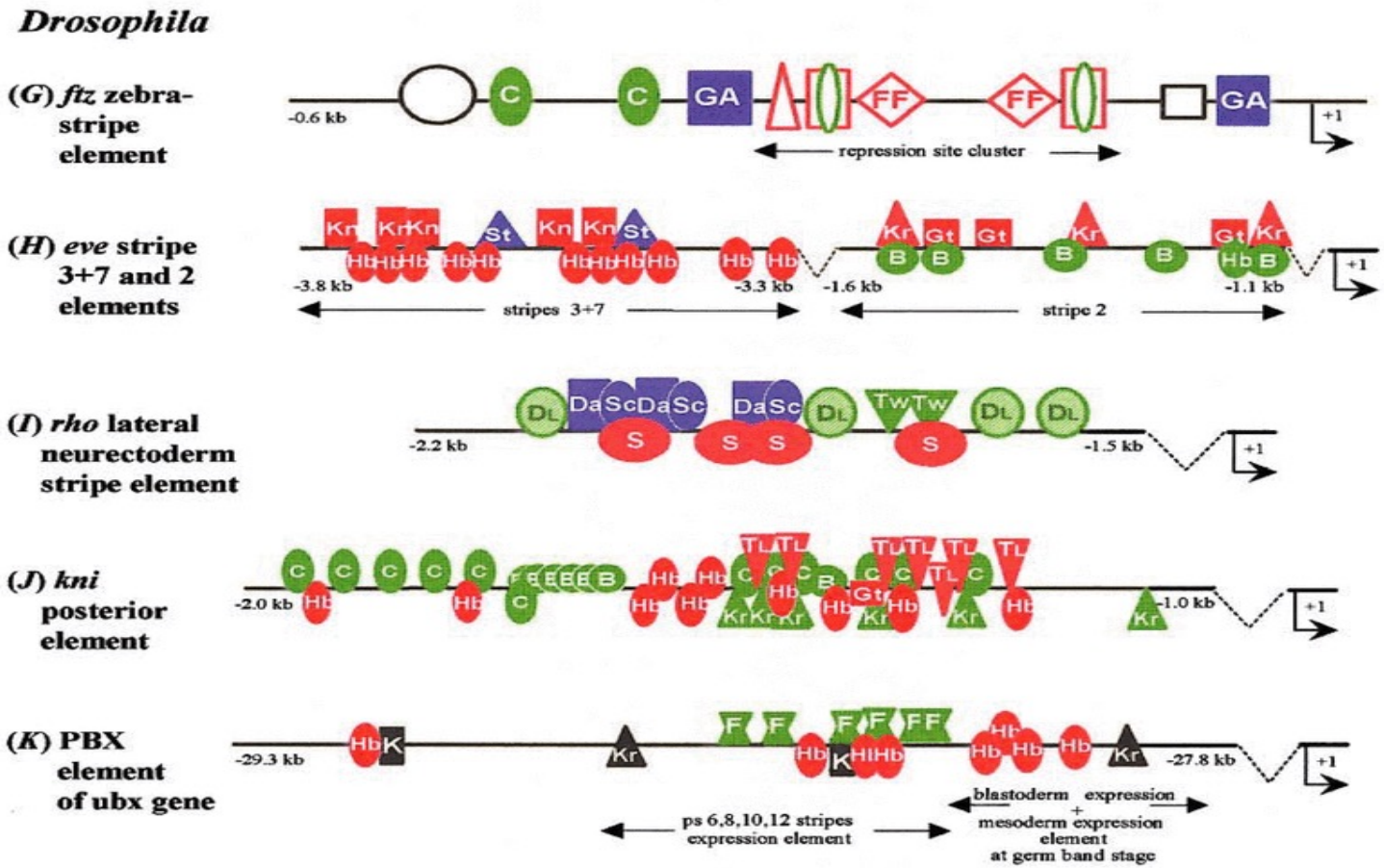
PhyloGibbs runs on arbitrary collections of multiple local sequence alignments of orthologous sequences. The algorithm searches over all ways in which an arbitrary number of binding sites for an arbitrary number of transcription factors can be assigned to the multiple sequence alignments. These binding site configurations are scored by a Bayesian probabilistic model that treats aligned sequences by an explicit model for the evolution of binding sites and 'background' intergenic DNA that takes the phylogenetic relationship between the species in the alignment into account. The algorithm uses simulated annealing and Monte-Carlo Markov-chain sampling to rigorously assign posterior probabilities to all the binding sites that it reports.

List of the most important features:

- The algorithm can search for an arbitrary number of sites for an arbitrary number of different regulatory motifs. The user can either specify the total number of sites and motifs that PhyloGibbs needs to search for, or it can supply PhyloGibbs with a guess for the total number of sites and motifs in the data.
- The algorithm rigorously takes into account the phylogenetic relationships of the species from which the input data derive. This allows PhyloGibbs to distinguish conservation that is due to the occurrence of functional sites from spurious conservation that is due to the evolutionary proximity of the species. Example phylogenetic trees for commonly used species can be downloaded from the download page.
- PhyloGibbs uses an anneal+track strategy that rigorously assigns posterior probabilities to the sites it reports. In the anneal stage the globally maximum-a-posterior-probability set of binding sites is identified and their posterior probabilities are calculated in the tracking stage.
- The program can also be used to calculate the statistical significance of a pre-specified set of putative binding sites.
- Background probabilities for nonfunctional sequences are implemented as Markov models of arbitrary order (to be specified by the user). Background models can be calibrated from externally supplied files with background sequences.
- Users can specify informative priors for the motifs by supplying an external file with weight matrices. This allows the algorithm to automatically identify new binding sites for motifs for qstawhich one or more binding sites are already known.

Final question: Discovery of enhancer modules

Clusters of binding sites



(from Arnone, M. I. and Davidson, E. H., *Development,* **124**(10):1851-64, 1997.)

Ahab scoring model

Rajewsky et al. BMC Bioinformatics 2002

Rather than assuming that we expect binding sites to be uniformly distributed across the sequence, Ahab assumes that different regions of the upstream sequence have *different* prior probabilities of containing binding sites.

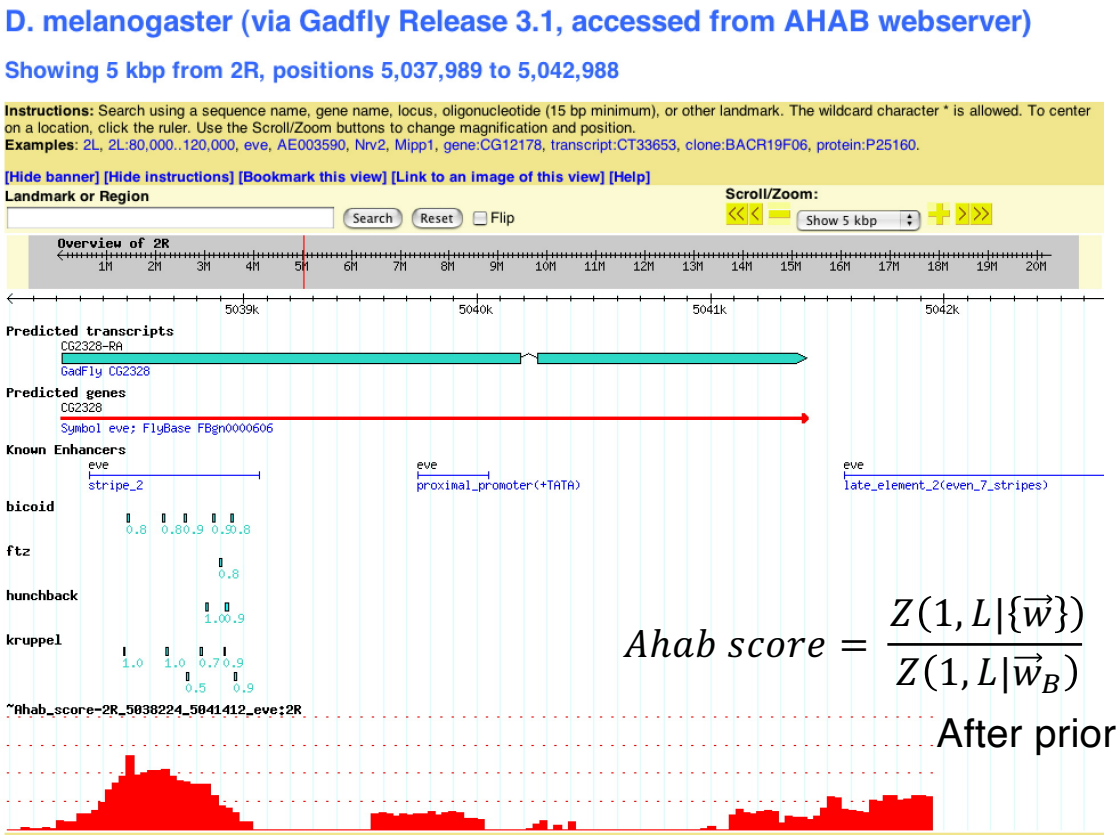
Thus, the likelihood of a sequence segment given a weightmatrix depends not only on the sequence, but also on the priorprobability of a site for the transcription factor in that genomic region:

$$P(s[i..i+\omega-1|\vec{w}) = \pi \prod_{j=1}^{\omega} w^{j}_{s[i+j]}$$

Ahab determines the set of priors, for all of the ***m*** weight matrices that maximize the partition function *for a given region of the upstream sequence* (of length typical for enhancer elements).

Discovering enhancer modules

Ahab annotation of *eve* gene



$$Ahab\ score = \frac{Z(1, L|\{\vec{w}\})}{Z(1, L|\vec{w}_B)}$$

After prior optimization