Probabilistic models of biological sequence motifs

Description of Known Motifs

AGB - Master in Bioinformatics UPF

2015-2016

Eduardo Eyras

Computational Genomics

Pompeu Fabra University - ICREA

Barcelona, Spain

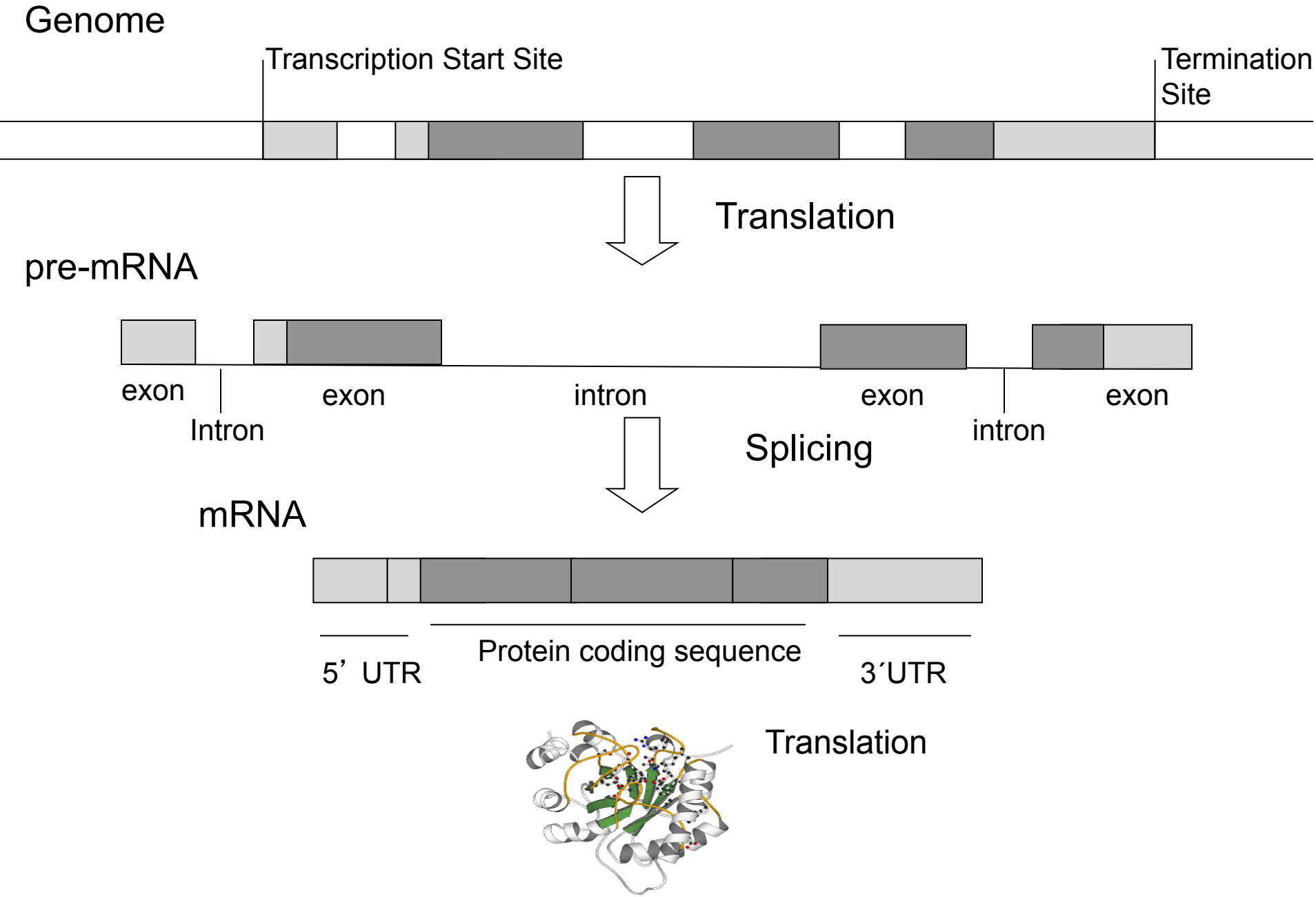
What we will see

How to build simple probabilistic models to describe sequence motifs (using a training set).

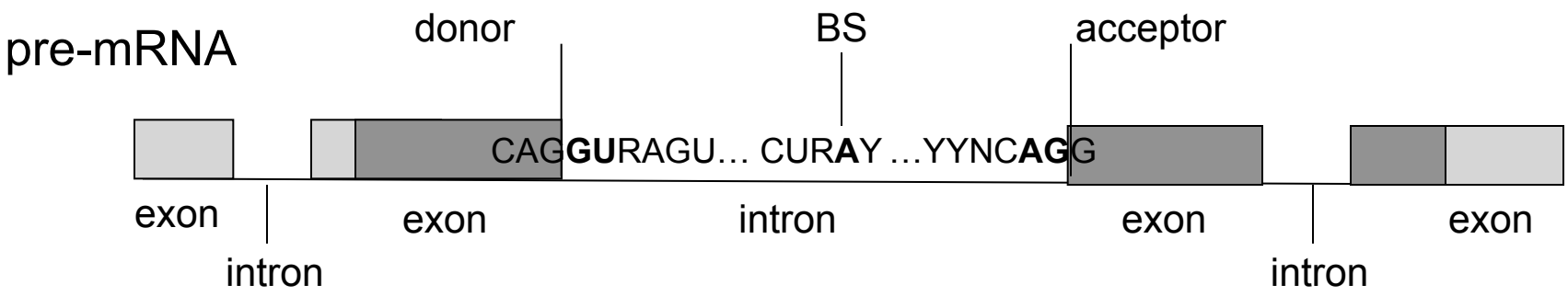
How to study the motif properties in terms of heterogeneity and dependencies between positions

How to model dependencies between positions.

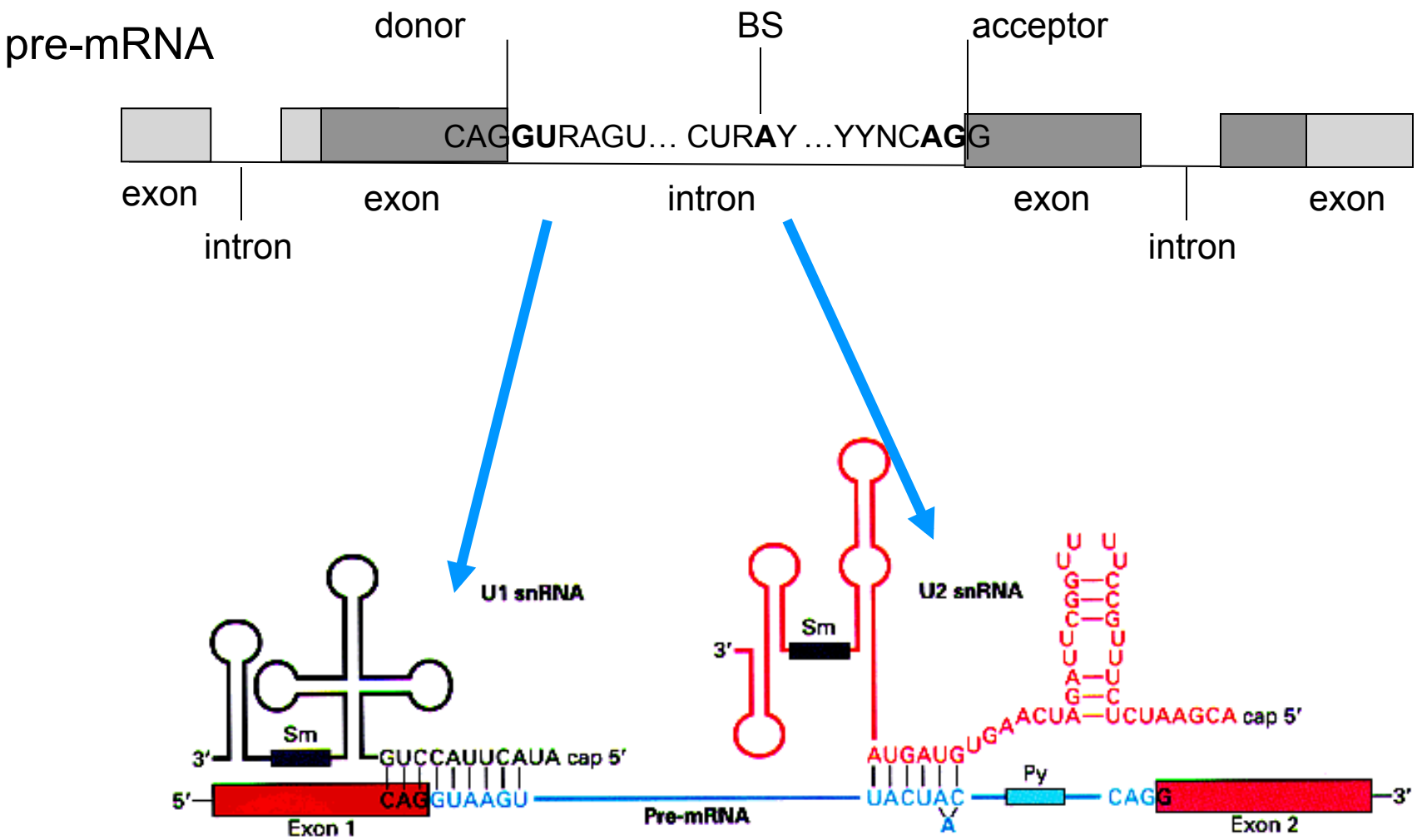
Added complexity: RNA processing (e.g. Splicing)



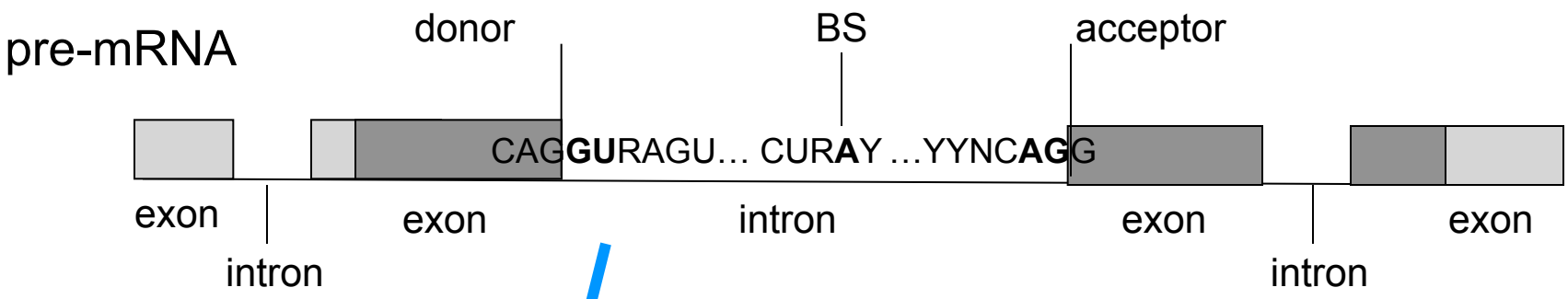
Splice-site signals

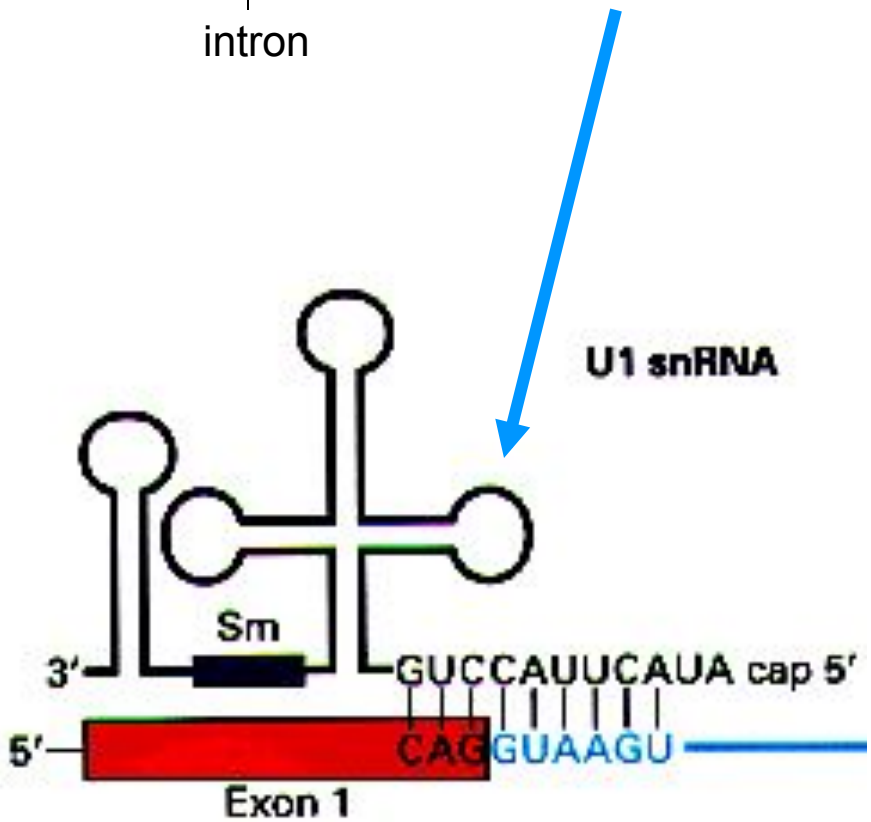


Splice-site signals



Splice-site signals





Description of signals (motifs)

Exact word

1 example

CAG**GT**AAGT

Consensus

Multiple examples

CAGGTAAGT
TAGGTGAGC
GTAGTAAGA
CAAGTAATA
ATGGTAATG
CAGGTGATC
AAGGTGAGC

NWR**GT**RAKN Consensus motif

Summary of single-letter code recommendations

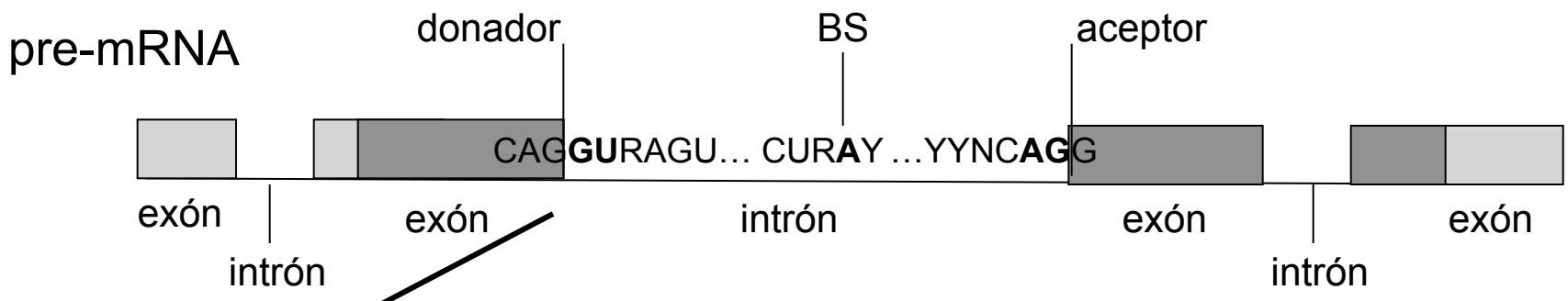
Symbol	Meaning	Origin of designation
G	G	Guanine
A	A	Adenine
T	T	Thymine
C	C	Cytosine
R	G or A	puRine
Y	T or C	pYrimidine
M	A or C	aMino
K	G or T	Keto
S	G or C	Strong interaction (3 H bonds)
W	A or T	Weak interaction (2 H bonds)
H	A or C or T	not-G, H follows G in the alphabet
B	G or T or C	not-A, B follows A
V	G or C or A	not-T (not-U), V follows U
D	G or A or T	not-C, D follows C
N	G or A or T or C	aNy

The simplest probabilistic model

Position Weight Matrix (PWM)

(position specific scoring matrix (PSSM))

Weight Matrices




```
----exon----intron
    CAGGTACCC
    GAGGTGAGA
    CTGGTGAGG
    TAGGTGAGT
    CAGGTCTGT
    CTGGTGAGC
    CAGGTAAGT
```

E.g. position 1, P(C) = frequency = 5/7 = 0.71

pos	1	2	3	4	5	6	7	8	9
A	0	0.71	0	0	0	0.28	0.71	0	0.14
C	0.71	0	0.28	0	0	0.14	0.14	0.14	0.28
G	0.14	0	0.71	1	0	0.57	0	0.85	0.14
T	0.14	0.28	0	0	1	0	0.14	0	0.42

Observations (real splice-sites)

Testing for a new functional site

pos	1	2	3	4	5	6	7	8	9
A	0	0.71	0	0	0	0.28	0.71	0	0.14
C	0.71	0	0.28	0	0	0.14	0.14	0.14	0.28
G	0.14	0	0.71	1	0	0.57	0	0.85	0.14
T	0.14	0.28	0	0	1	0	0.14	0	0.42

What is the probability that a sequence contains a functional site described by this model

$$S = s_1 s_2 s_3 \ldots s_n$$

We can calculate the probability that *S* is given by the model obtained from the observations:

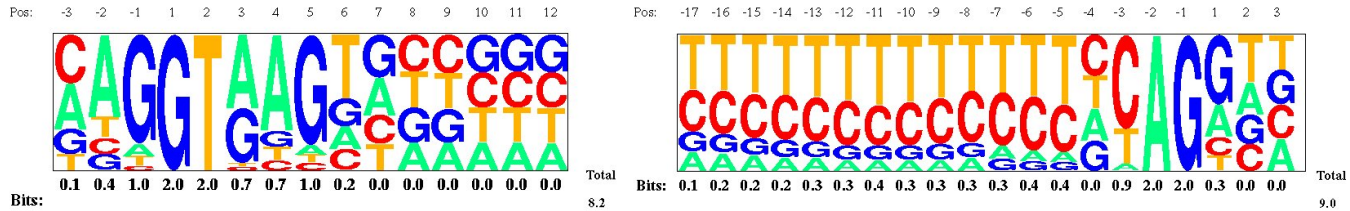
$$P(S) = P(s_1 s_2 \ldots s_N) = P(s_1, pos = 1) P(s_2, pos = 2) \ldots P(s_N, pos = n)$$

Implicitly, we assume independence between adjacent positions

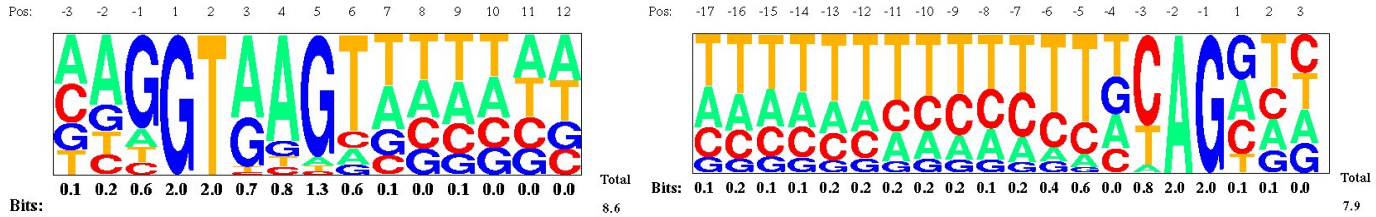
Graphical Representation: Sequence Logos

pos	1	2	3	4	5	6	7	8	9
A	0	0.71	0	0	0	0.28	0.71	0	0.14
C	0.71	0	0.28	0	0	0.14	0.14	0.14	0.28
G	0.14	0	0.71	1	0	0.57	0	0.85	0.14
T	0.14	0.28	0	0	1	0	0.14	0	0.42

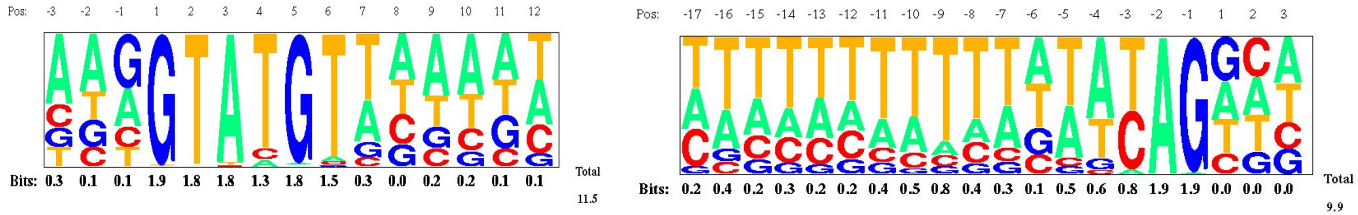
human



drosophila



yeast



http://weblogo.berkeley.edu/logo.cgi

Pseudocounts

In any observed data set there is the possibility, especially with low-probability events and/or **small data sets**, of a possible event not occurring.

Its observed frequency is therefore 0, implying a probability of 0.

```
-exon--intron-
CAGGTACCC
GAGGTGAGA
CTGGTGAGG
TAGGTGAGT
CAGGTCTGT
CTGGTGAGC
CAGGTAAGT
```

Estimated probability P(A, pos=1) = 0

We may wrongly infer that lack of A is characteristic of splice-sites (overfitting)

Simplest solution: modify the counting:
n_i $n_i + p$, i=1,2,3,4. E.g.:

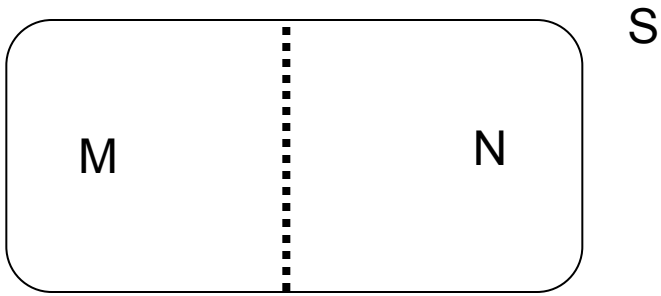
$$P(A) = \frac{n_A + mp}{n_A + n_C + n_G + n_T + m}$$

Laplace rule: pseudocount *m=4, p=1/4*

Hypothesis testing

Problem: choosing between two models M, N to represent a data set

Each model represents a prob distribution of the sample space S



We need Statistical test that can distinguish between the two models

To distinguish between two models, we consider the Likelihood-ratio between two possible models:

$P(s \mid M) = P(s)$ under M

$P(s \mid N) = P(s)$ under N

$$LR = \frac{P(S \mid M)}{P(S \mid R)} = \frac{P(s_1, pos = 1 \mid M)}{P(s_1, pos = 1 \mid R)} \cdots \frac{P(s_N, pos = n \mid M)}{P(s_N, pos = n \mid R)}$$

Likelihood ratio

In general, we want to compare the model of real sites *M* with an alternative (false site) model *R*

Example of alternative models:

- Random sequences (P(a)=0.25, for a=A,C,G,T)
- False sites (sequences with GT but are not real donors)

$$LR = \frac{P(S \mid M)}{P(S \mid R)} = \frac{P(s_1, pos = 1 \mid M)}{P(s_1, pos = 1 \mid R)} \cdots \frac{P(s_N, pos = n \mid M)}{P(s_N, pos = n \mid R)}$$

pos	1	2	3	4	5	6	7	8	9
A	0	2.84	0	0	0	1.12	2.84	0	0.56
C	2.84	0	1.12	0	0	0.56	0.56	0.56	1.12
G	0.56	0	2.84	4	0	2.28	0	3.4	0.56
T	0.56	1.12	0	0	4	0	0.56	0	1.68

Likelihood ratio

In general, we want to compare the model of real sites *M* with an alternative (false site) model *R*

Example of alternative models:

- Random sequences (P(a)=0.25, for a=A,C,G,T)
- False sites (sequences with GT but are not real donors)

$$LR = \frac{P(S \mid M)}{P(S \mid R)} = \frac{P(s_1, pos = 1 \mid M)}{P(s_1, pos = 1 \mid R)} \cdots \frac{P(s_N, pos = n \mid M)}{P(s_N, pos = n \mid R)}$$

pos	1	2	3	4	5	6	7	8	9
A	0	2.84	0	0	0	1.12	2.84	0	0.56
C	2.84	0	1.12	0	0	0.56	0.56	0.56	1.12
G	0.56	0	2.84	4	0	2.28	0	3.4	0.56
T	0.56	1.12	0	0	4	0	0.56	0	1.68

Using uniform background

Position Weight Matrices (PWMs)

Probabilities are small

Problem multiplying probabilities (too small to be correctly handled by a computer)
Solution: use logarithms:

$M_{ai} =$

pos	1	2	3	4	5	6	7	8	9
A	-999	1.02	-999	-999	-999	-0.22	1.02	-999	-0.91
C	1.02	-999	-0.22	-999	-999	-0.91	-0.91	-0.91	-0.22
G	-0.91	-999	1.02	1.38	-999	0.69	-999	1.16	-0.91
T	-0.91	-0.22	-999	-999	1.38	-999	-0.91	-999	0.47

Log-Likelihood ratio

$$\log LR = \log \frac{P(s_1 \mid M)P(s_2 \mid M)...P(s_n \mid M)}{P(s_1 \mid R)P(s_2 \mid R)...P(s_n \mid R)} = \sum_{i=1}^{n} \log \frac{P(s_i \mid M)}{P(s_i \mid R)}$$

Log 0 is generally set up to be a large negative number
THe alternative is to use pseudocounts

Position Weight Matrices (PWMs)

Consider the example: CTGGTAAGC

$M_{a,i} =$

pos	1	2	3	4	5	6	7	8	9
A	-999	1.02	-999	-999	-999	-0.22	1.02	-999	-0.91
C	1.02	-999	-0.22	-999	-999	-0.91	-0.91	-0.91	-0.22
G	-0.91	-999	1.02	1.38	-999	0.69	-999	1.16	-0.91
T	-0.91	-0.22	-999	-999	1.38	-999	-0.91	-999	0.47

$$\log L = \log \frac{P(CTGGTAAGC \mid M)}{P(CTGGTAAGC \mid R)}$$

$$= \log \frac{P_1(C \mid M)P_2(T \mid M)P_3(G \mid M) \cdot \ldots \cdot P_8(G \mid M)P_9(C \mid M)}{P_1(C \mid R)P_2(T \mid R)P_3(G \mid R) \cdot \ldots \cdot P_8(G \mid R)P_9(C \mid R)}$$

$$= \log \frac{P_1(C \mid M)}{P_1(C \mid R)} + \log \frac{P_2(T \mid M)}{P_2(T \mid R)} + \log \frac{P_3(G \mid M)}{P_3(G \mid R)} + \ldots + \log \frac{P_8(G \mid M)}{P_8(G \mid R)} + \log \frac{P_9(C \mid M)}{P_9(C \mid R)}$$

$$= M_{C,1} + M_{T,2} + M_{G,3} + \ldots + M_{G,8} + M_{C,9}$$

$$= 1.02 - 0.22 + 1.02 + 1.38 + 1.38 - 0.22 + 1.02 + 1.16 - 0.22$$

$$= 6.32$$

Position Weight Matrices (PWMs)

To search in an unknown sequence for this motif (for the possibility that there is a donor splice site), we use a **sliding window** along the sequence of the same size as the motif, and at each position we score the similarity to the motif using the score given by the Matrix. sites:

```
..CGTGAGTCGGGGTGAGAGCATGCTGGTAAGCCCGGCTGGTGAACTGCCGGTAGTC..
```

Assign score to each 9 base window. Use score cutoff to predict potential 5' splice sites

$$\log LR = \log \frac{P(S \mid M)}{P(S \mid R)} = \log \frac{P(s_1, pos = 1 \mid M)}{P(s_1, pos = 1 \mid R)} + \ldots + \log \frac{P(s_N, pos = n \mid M)}{P(s_N, pos = n \mid R)} > a$$

log *LR* > a => it is more likely to correspond to a case of model M
log *LR* < a => it is more likely to correspond to a case of model R

Position Weight Matrices (PWMs)

Since GT is invariable, we could skip the contribution of positions 4 and 5, and consider only query sequences with GT at these positions

$M_{ai} =$

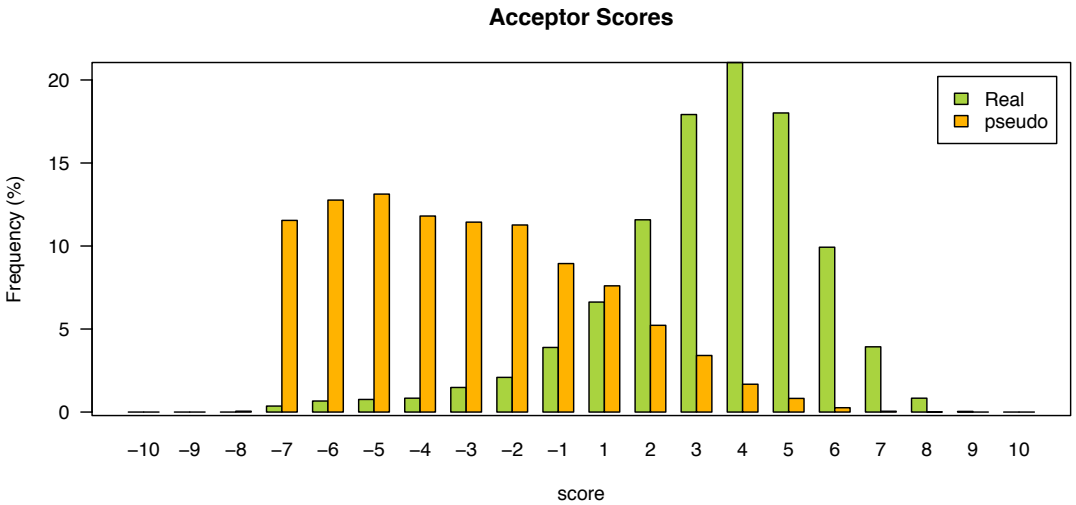
pos	1	2	3	4	5	6	7	8	9
A	-999	1.02	-999	-999	-999	-0.22	1.02	-999	-0.91
C	1.02	-999	-0.22	-999	-999	-0.91	-0.91	-0.91	-0.22
G	-0.91	-999	1.02	1.38	-999	0.69	-999	1.16	-0.91
T	-0.91	-0.22	-999	-999	1.38	-999	-0.91	-999	0.47

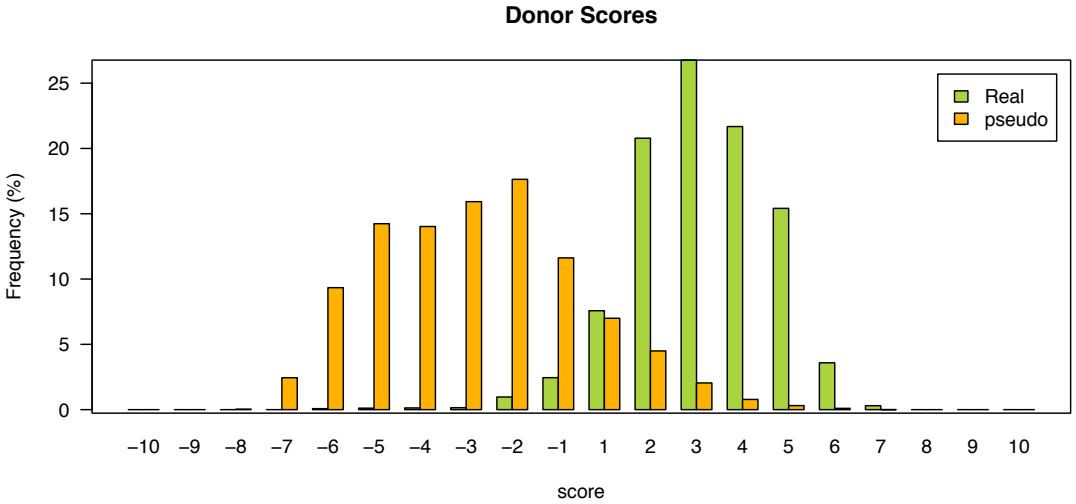
Example:

..CGTGAGTCGGGGTGAGAGCATGCTGGTAAGCCCGGCTGGTGAACTGCCGGTAGTC..

We would score only window1 and window2

Searching for motifs in novel sequences
(The sliding window approach)



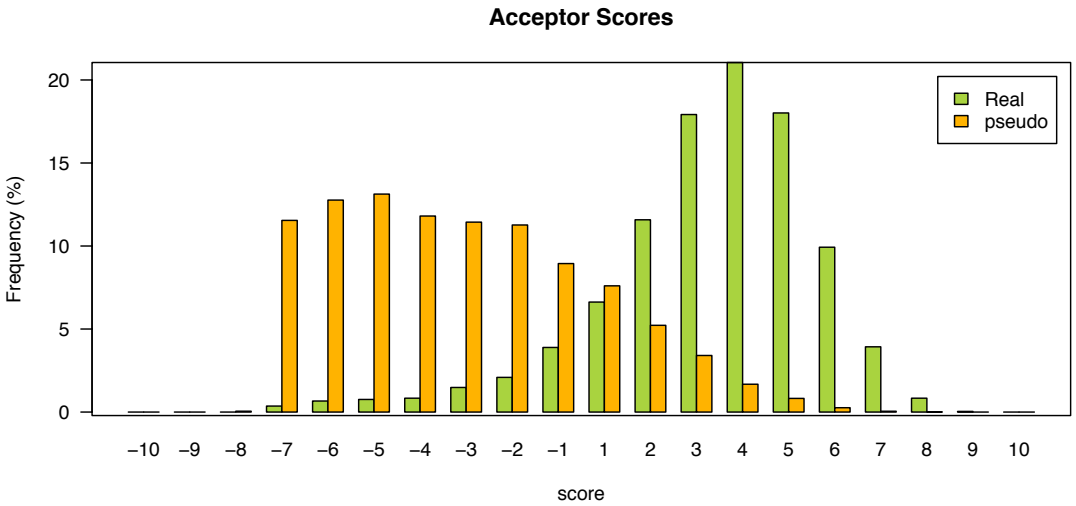


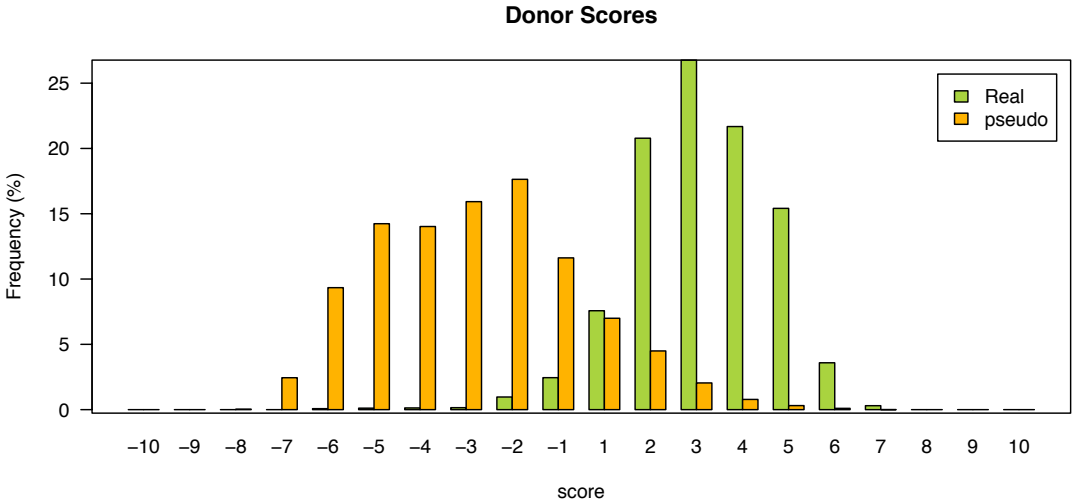
Compare Log-likelihood scores of real vs. pseudo splice sites

Training: on the training set, we calculate the cutoff for the desired Sn and Sp values

Testing: we use test data to evaluate the accuracy of our model

Searching for motifs in novel sequences
(The sliding window approach)





Compare Log-likelihood scores of real vs. pseudo splice sites

Recall:

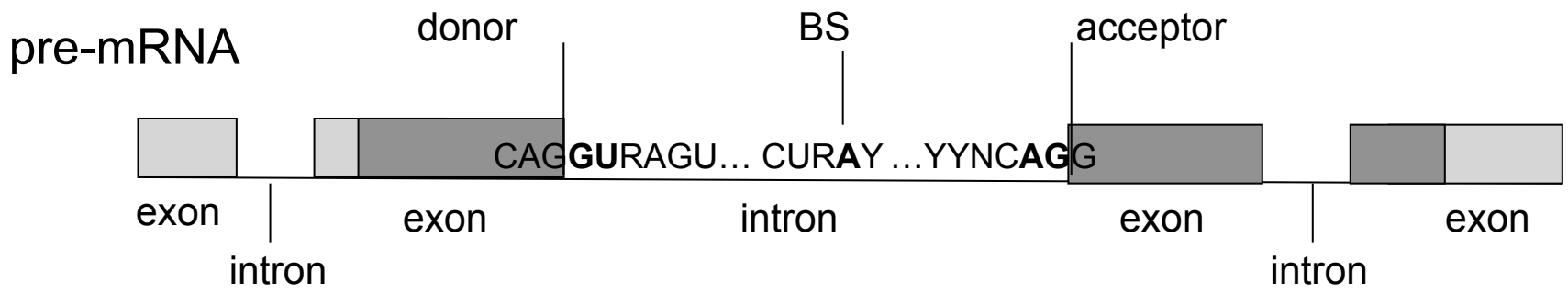
Sensitivity
fraction of real sites with score above cutoff

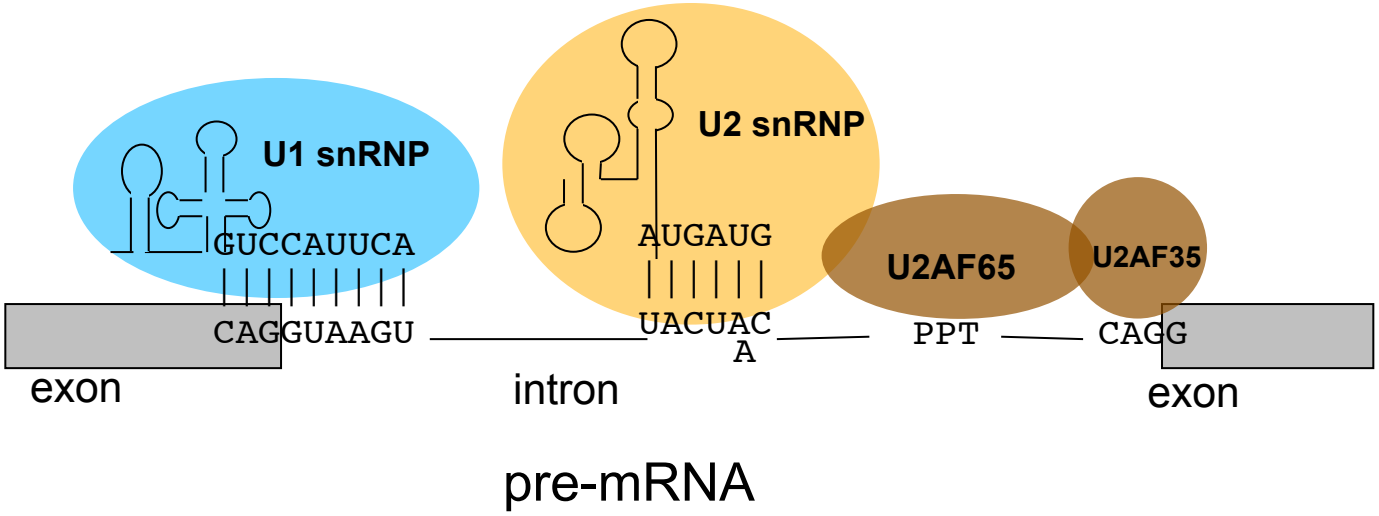
PPV
fraction of sites with score > cutoff that are true sites.

FPR
Fraction of negative cases (pseudo splice-sites) that are above the score cut-off

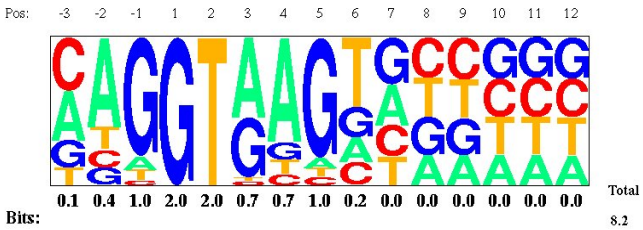
Determining the relevant positions

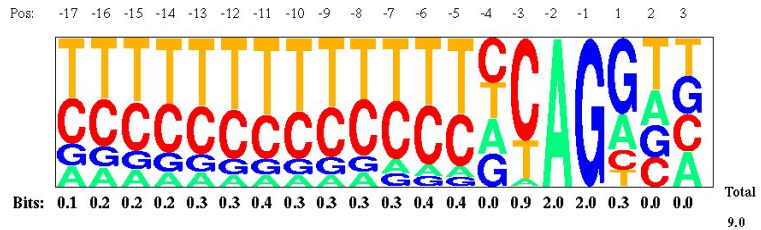
Splice-site signals

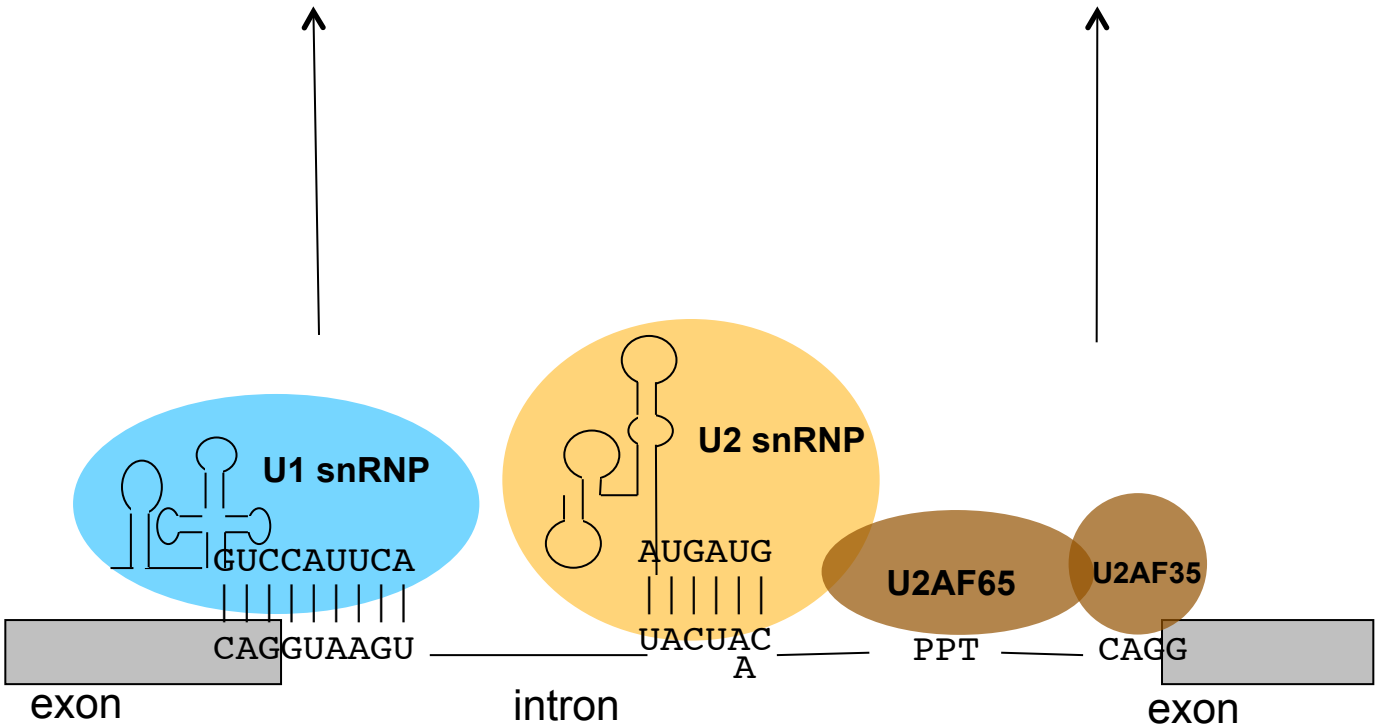




Splice-site signals

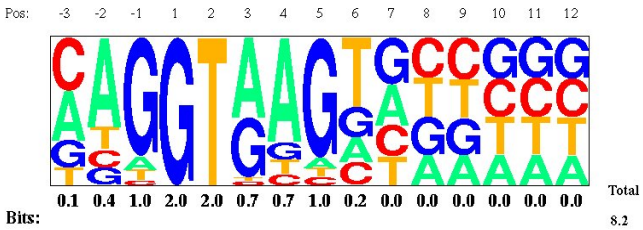


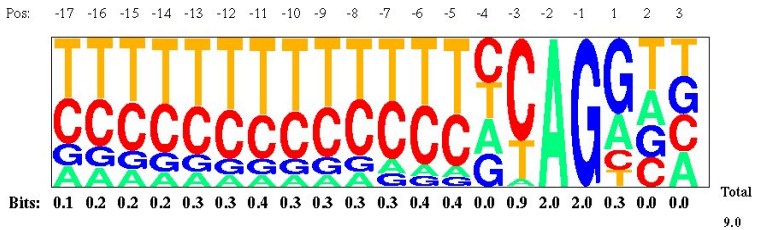




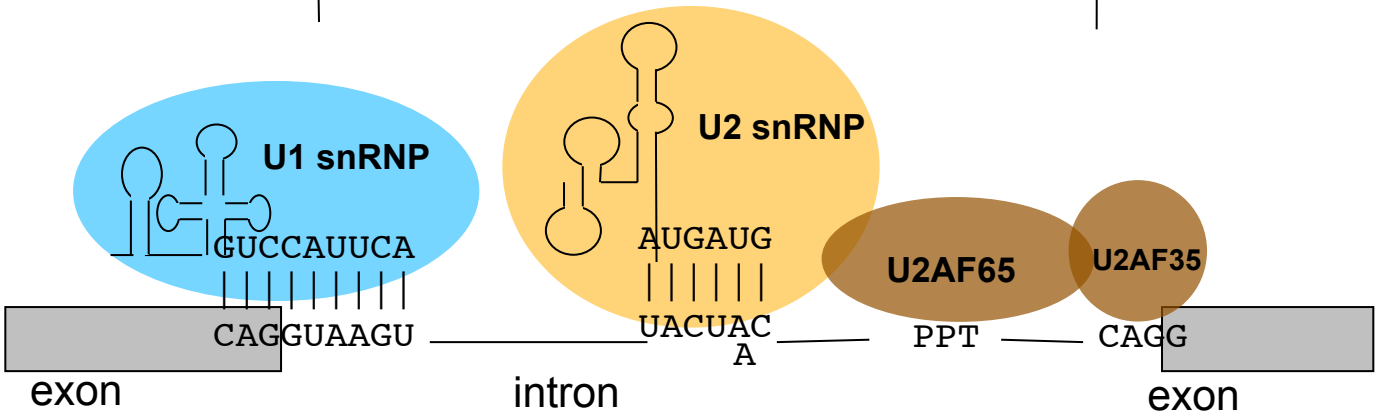
pre-mRNA

Splice-site signals



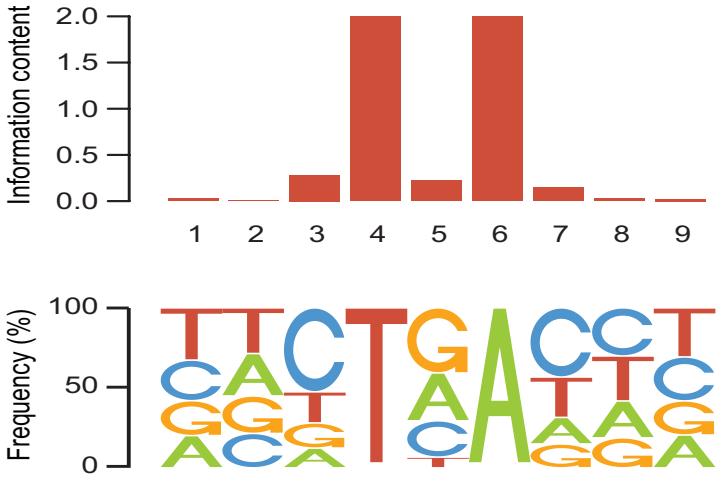


How many positions are relevant to model?



pre-mRNA

Information Content

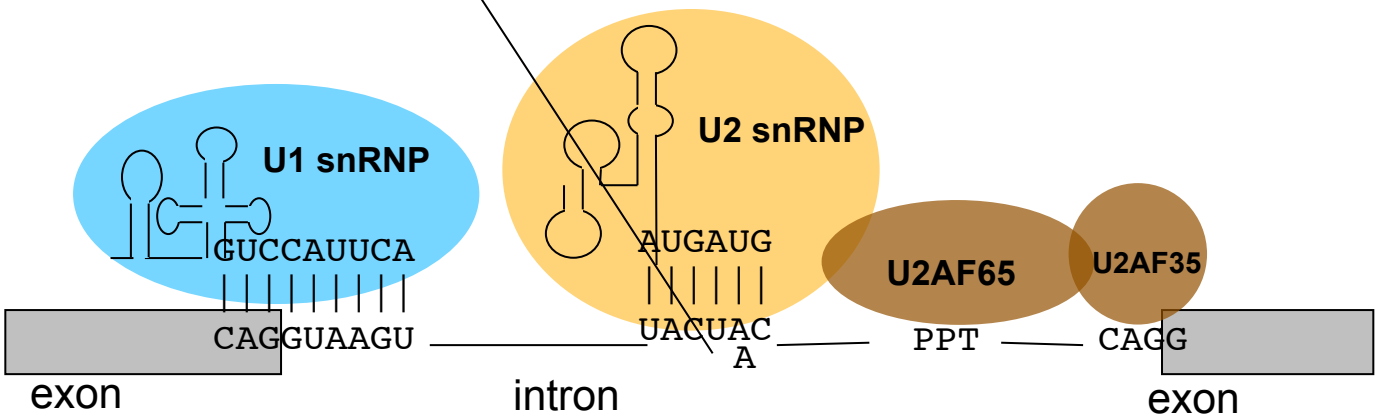


Frequencies

Information content
(The change in entropy comparing the expected and the observed distribution)

$$I_c(X) = H_{before} - H_{after}$$

$$= \log_2 N + \sum_{i=1}^{N} P(x_i) \log P(x_i)$$

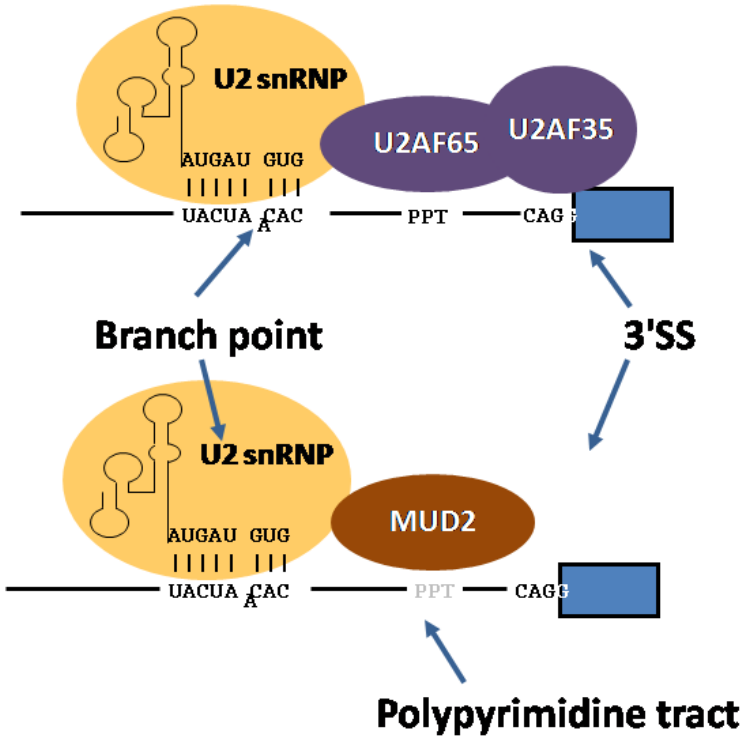


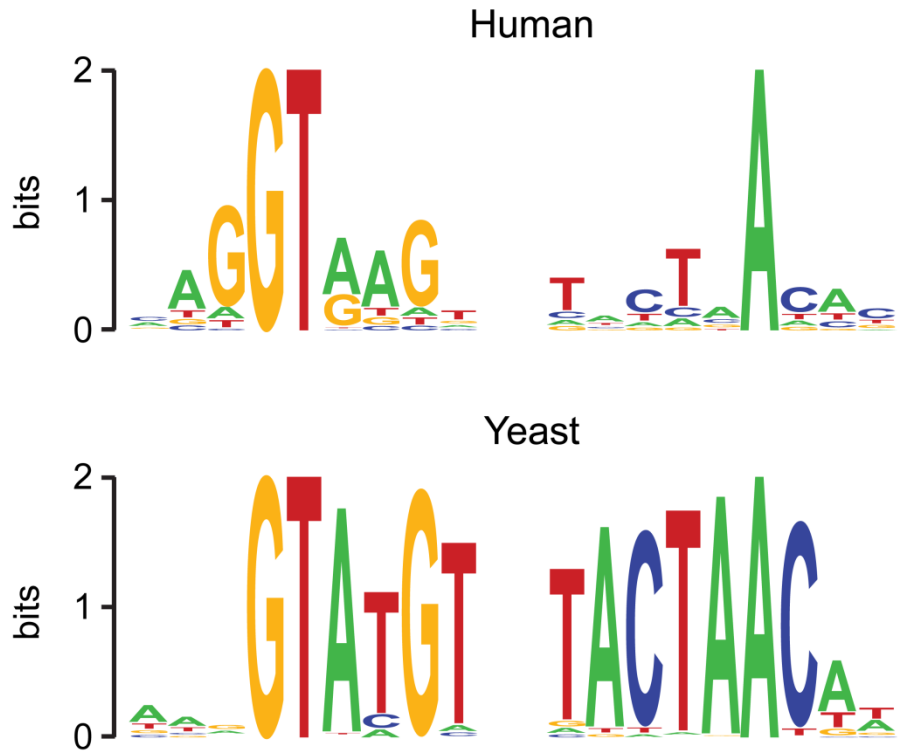
pre-mRNA

Corvelo et al. 2010

Information Content

H. sapiens





S. cerevisiae

Height is proportional to the Information content, the letter relative sizes are proportional to their frequencies (http://weblogo.berkeley.edu/logo.cgi)

Kullback-Leibler Divergence of two distributions

$$D(P \| Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$$

Also called the relative entropy, is the expected value of the log-rate of two distributions

$$D(P \| Q) = E(\log L) = \sum_{i=1}^{n} P(x_i) \log \frac{P(x_i)}{Q(x_i)} \qquad \log L = \log \frac{P(x)}{Q(x)}$$

The relative entropy is defined for two probability distributions that take values over the same alphabet (same symbols)

Kullback-Leibler Divergence of two distributions

$$D(P \,\|\, Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$$

The relative entropy is not a distance, but measures **how different two distributions are**

$D(P \,\|\, Q) \neq D(P \,\|\, Q)$ It is **not symmetric**

The value is never negative. It is zero when the 2 distributions are identical

$D(P \,\|\, Q) \geq 0$ with "=0 " for $P=Q$

The relative entropy provides a measure of the information content gained with the distribution *P* with respect to the distribution *Q*.

Its applications are **similar to those of the Information Content**

Better to apply *D* rather than *I* when background is not random

Exercise: (exam 2013)

Consider two discrete probability distributions *P* and *Q*, such that

$$\sum_i P(x_i) = 1 \quad \text{and} \quad \sum_i Q(x_i) = 1$$

Show that the relative entropy *D(P||Q)* is equivalent to the information content of *P* when the distribution *Q* is uniform.

Total Relative Entropy

To quantify the variability of an entire motif we can calculate the ***total*** relative entropy

P: distribution of the observed sequences corresponding to the motif
Q: distribution of a background model (e.g. Random sequences)

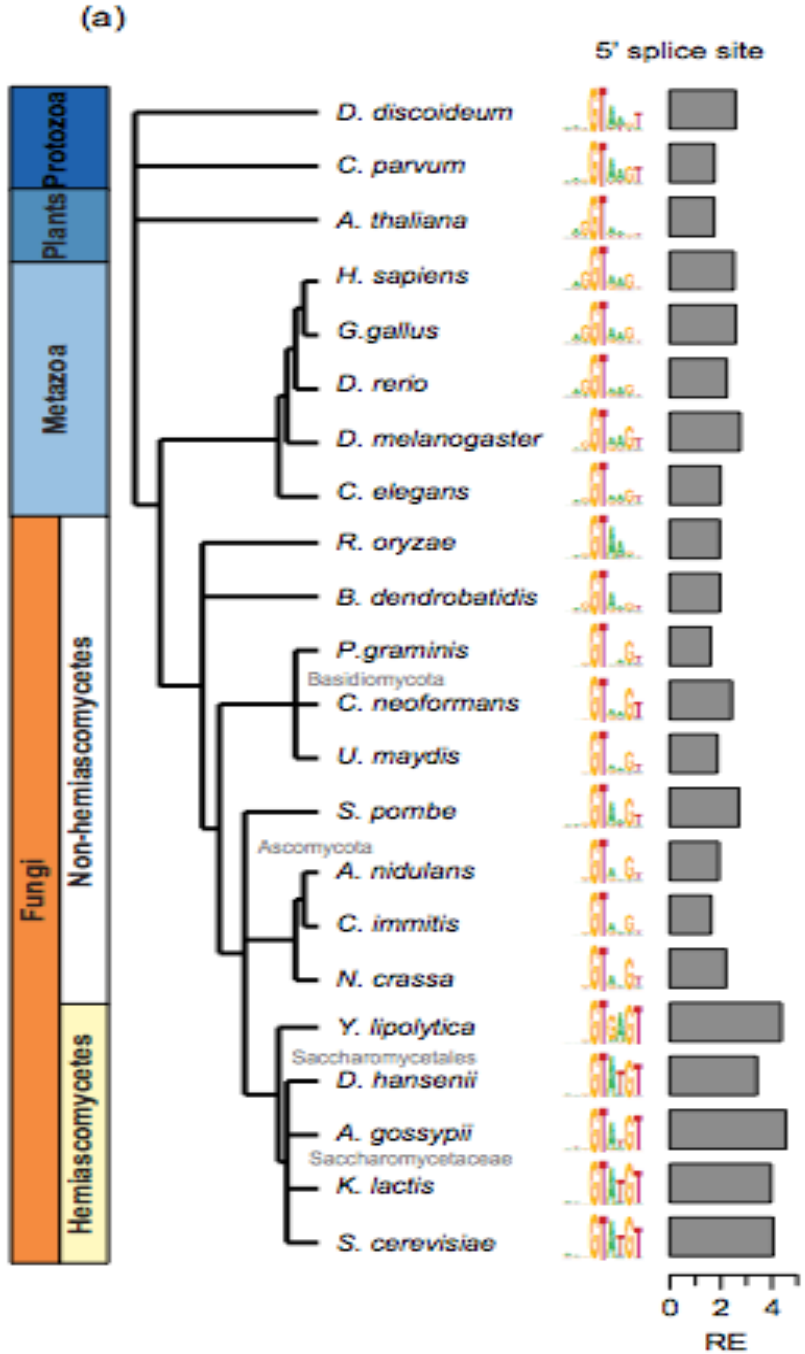
$$D(P \,\|\, Q) = \sum_{a=1}^{4} P(x_a) \log\left(\frac{P(x_a)}{Q(x_a)}\right)$$

Relative entropy at one position of the motif

$$D_{total}(P \,\|\, Q) = \sum_{i=1}^{N} \sum_{a=1}^{4} P(x_{a,i}) \log\left(\frac{P(x_{a,i})}{Q(x_{a,i})}\right)$$

Total Relative entropy

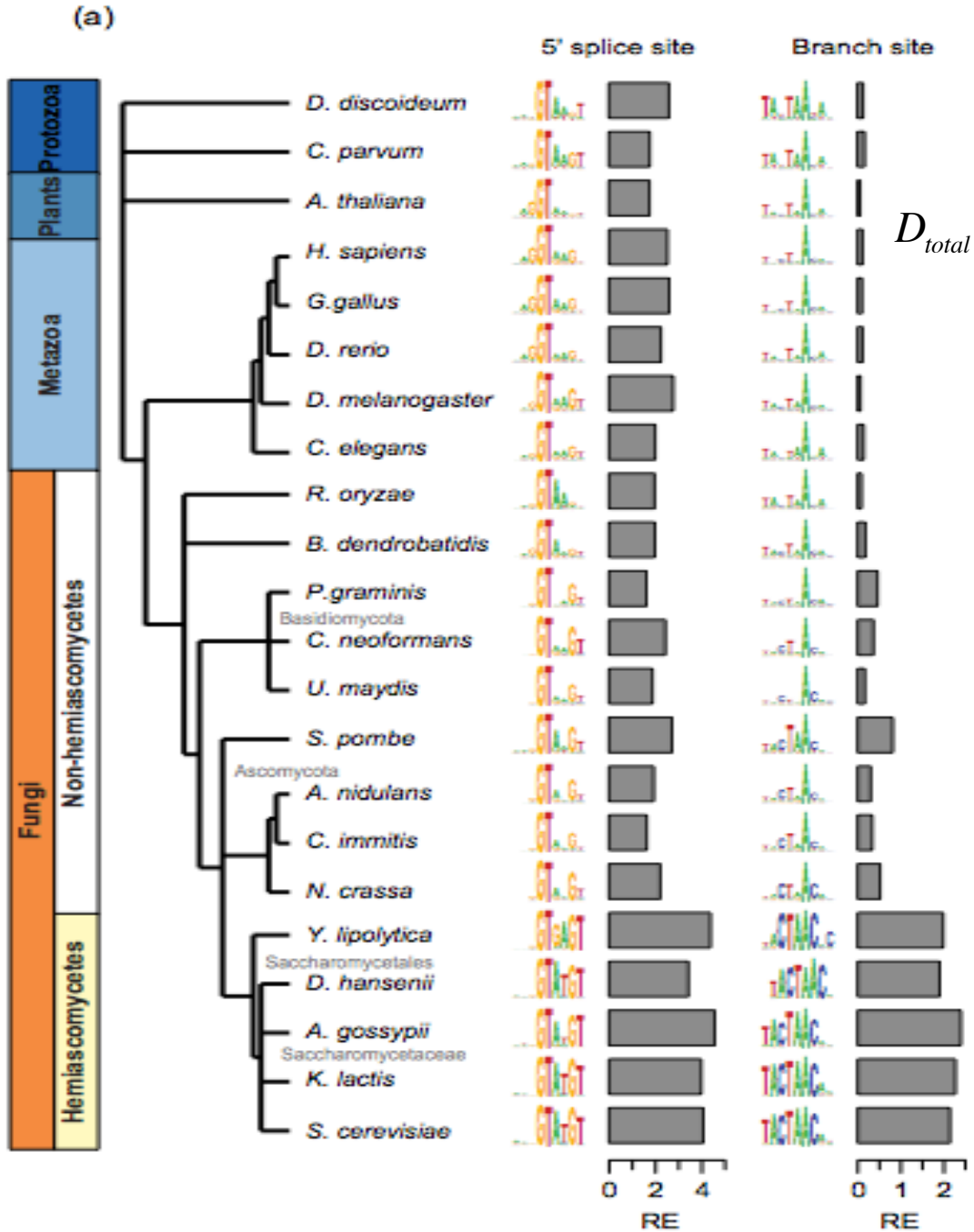
The total relative entropy *is calculated from the probability distribution of nucleotides, a=1,2,3,4, at each position, i=1,…,N, in the real signal,* $P(x_{a,i})$*, relative to the distribution in a* randomized set, $Q(x_{a,i})$*:*



Total Relative entropy

$$D_{total}(P,Q) = \sum_{i=1}^{N}\sum_{a=1}^{4} P(x_{a,i})\log\left(\frac{P(x_{a,i})}{Q(x_{a,i})}\right)$$

Plass et al. 2008



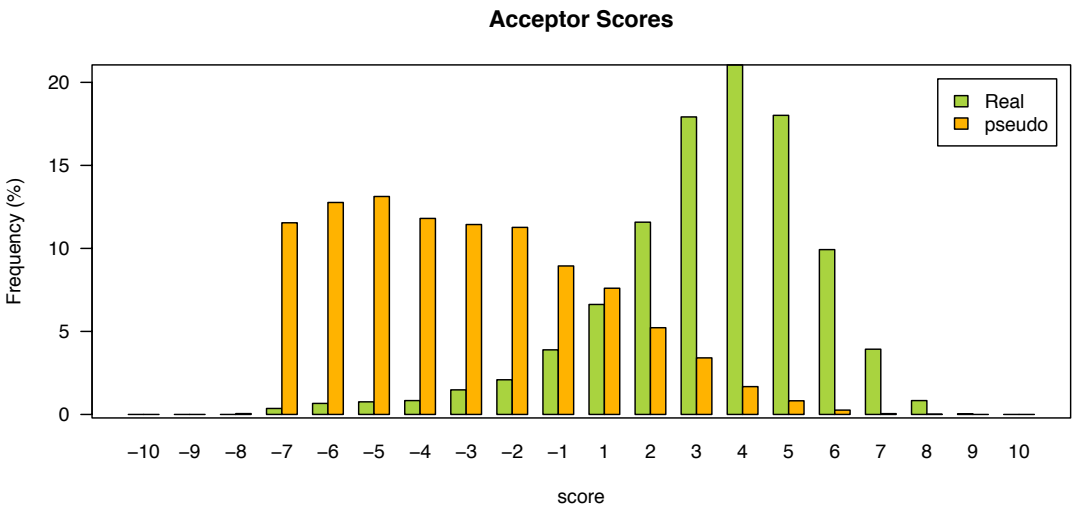
Total Relative entropy

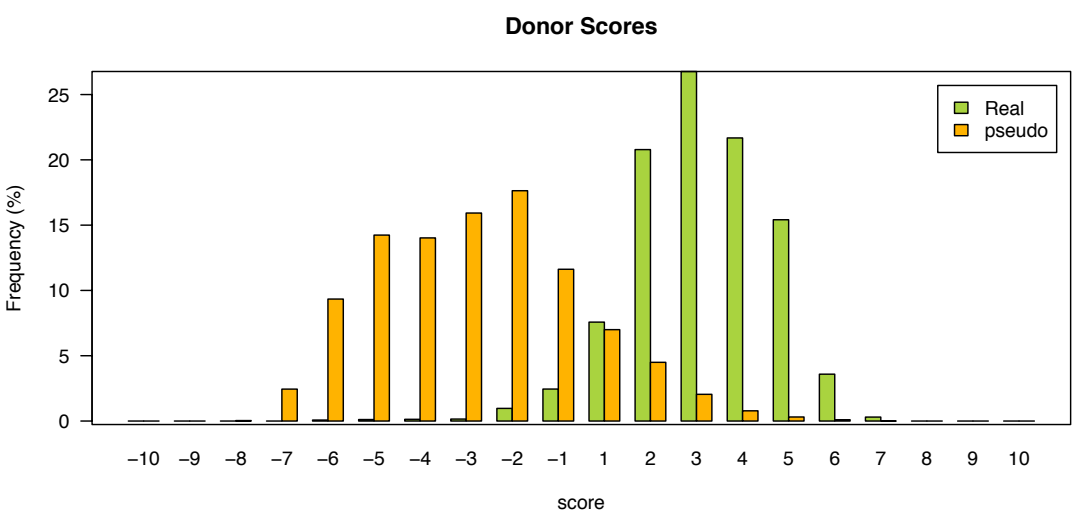
$$D_{total}(P,Q) = \sum_{i=1}^{N}\sum_{a=1}^{4} P(x_{a,i})\log\left(\frac{P(x_{a,i})}{Q(x_{a,i})}\right)$$

Plass et al. 2008

Modeling dependencies between positions

Modeling dependencies between positions





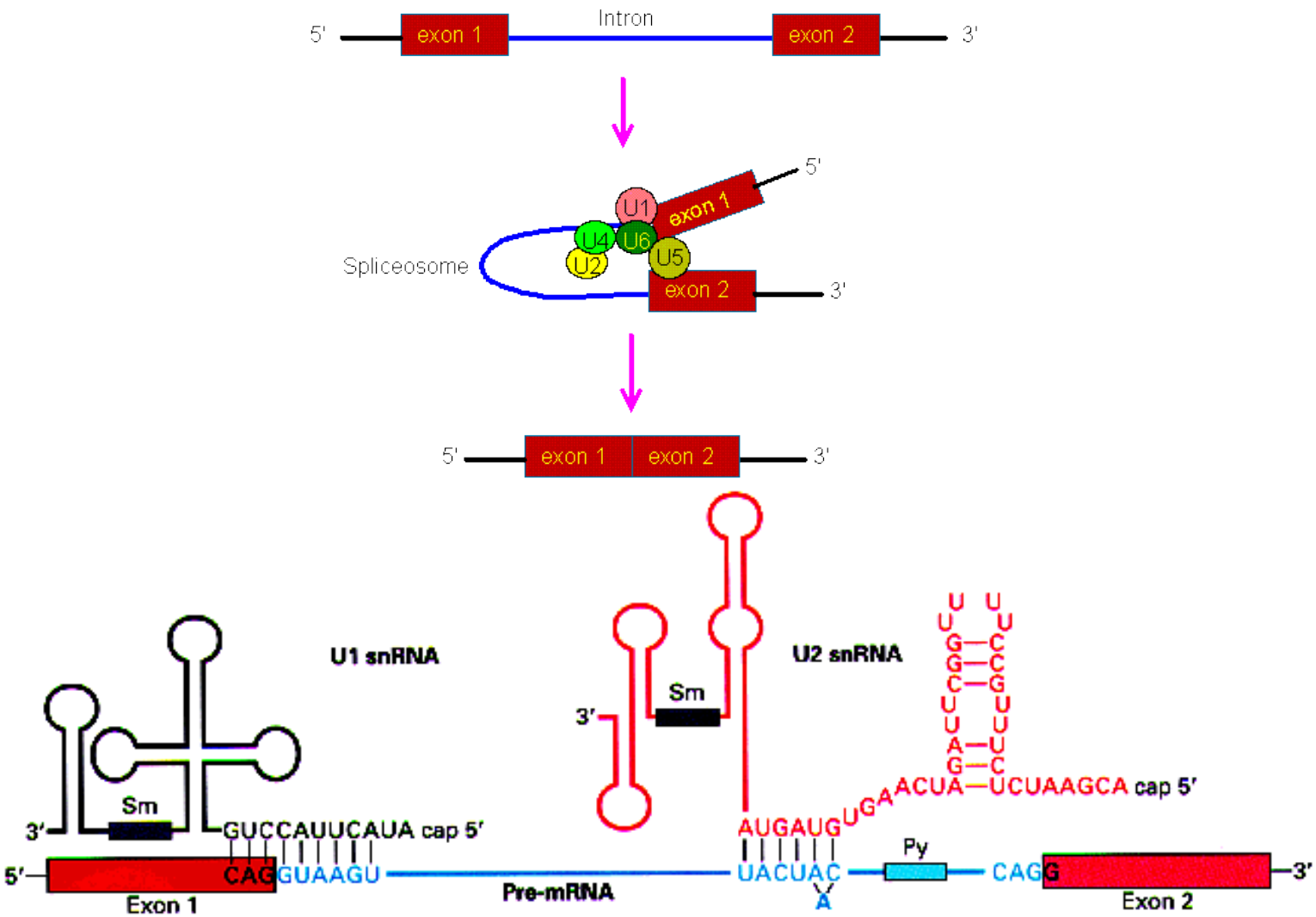
What does this result tell us?

A) Splicing machinery also uses other information besides 5' ss/3' ss motifs to identify splice sites

B) PWM model does not accurately capture some aspects of the 5' ss/ 3' ss that are used in recognition

C) Or both

Modeling dependencies between positions



Mutual information

Mutual Information

$$MI(X,Y) = H(X) + H(Y) - H(H,Y) = \sum_x \sum_y P(x,y) \log \frac{P(x,y)}{P(x)P(y)}$$

The mutual information of two random variables X and Y measures the dependencies between two variables, that is, information in X that is shared with Y.

e.g. X and Y take as values the nucleotides in two different positions, and the sum is carried out over the alphabet of nucleotides

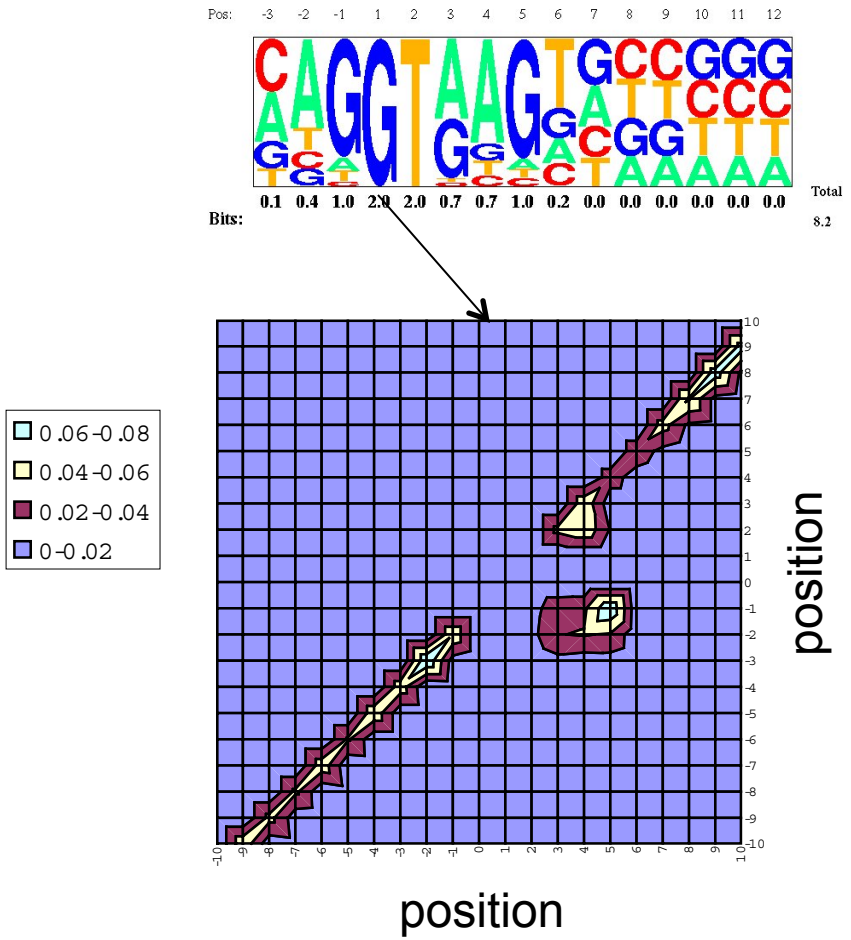
Independent positions *M(X,Y) = 0*

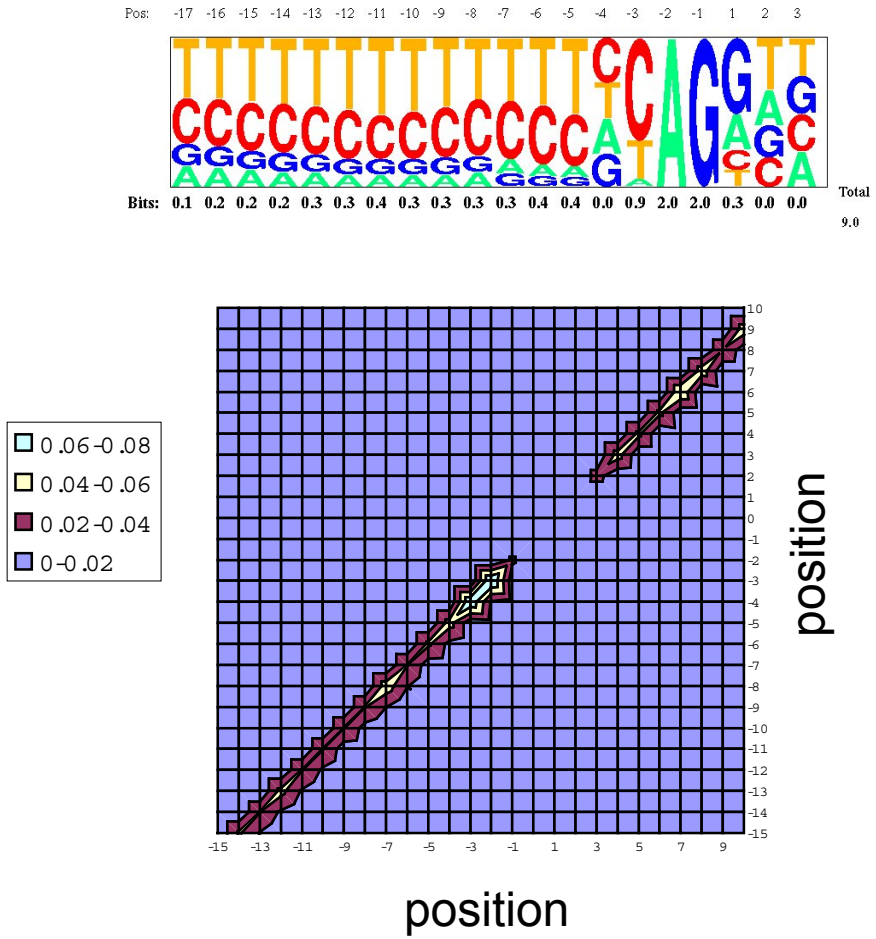
Dependent positions *M(X,Y)>0*

```
CTGAG
GTAGA
TTGAC
ATAGT
GTGAG
CTAAA
TTGAC
ATAAT

12345
```

Mutual information

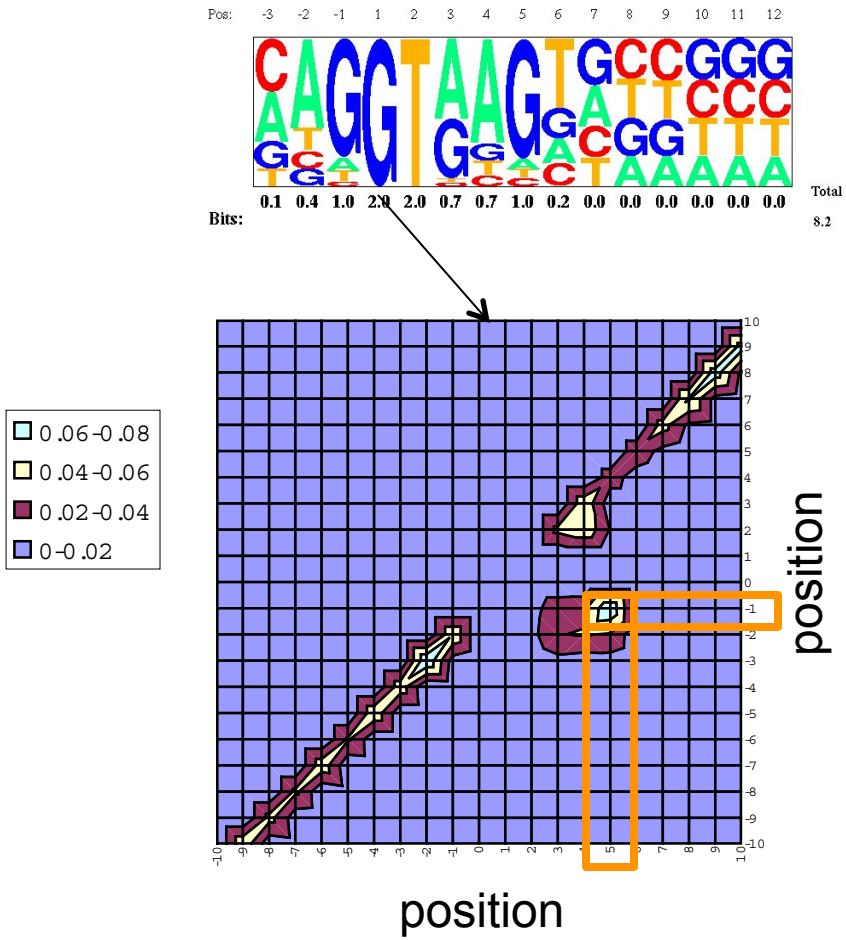


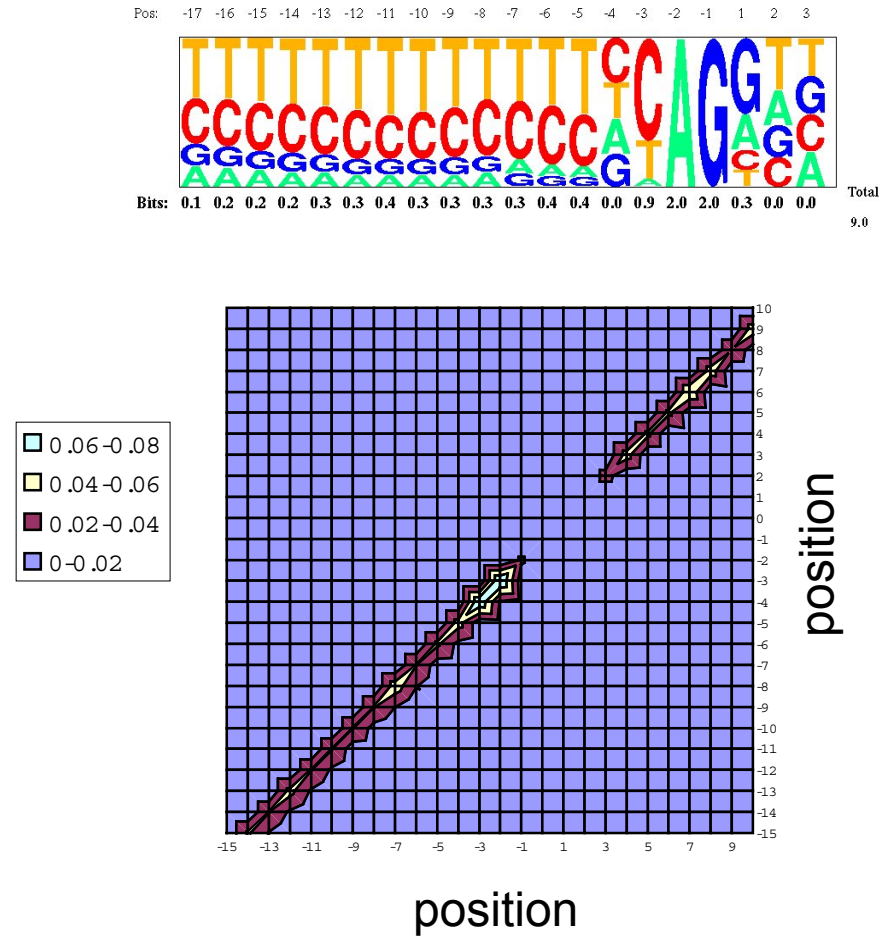


$$MI(X,Y) = \sum_{i=1}^{n} \sum_{j=1}^{n} P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i) P(y_j)}$$

Levine & Durbin 2001

Mutual information

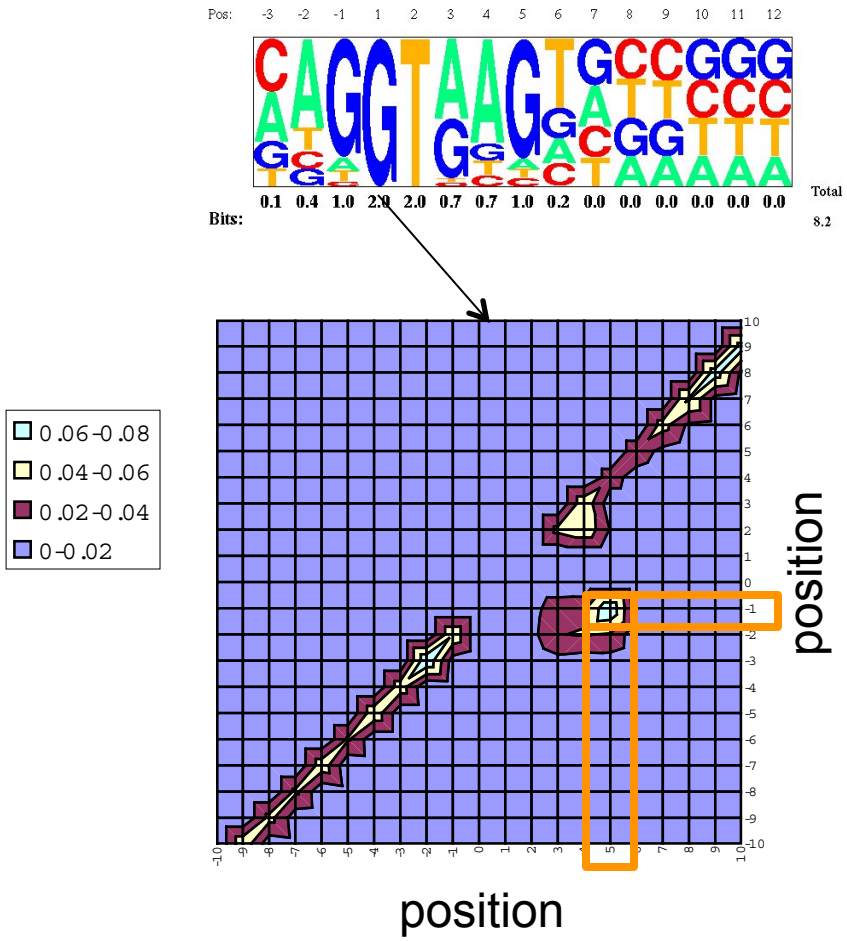


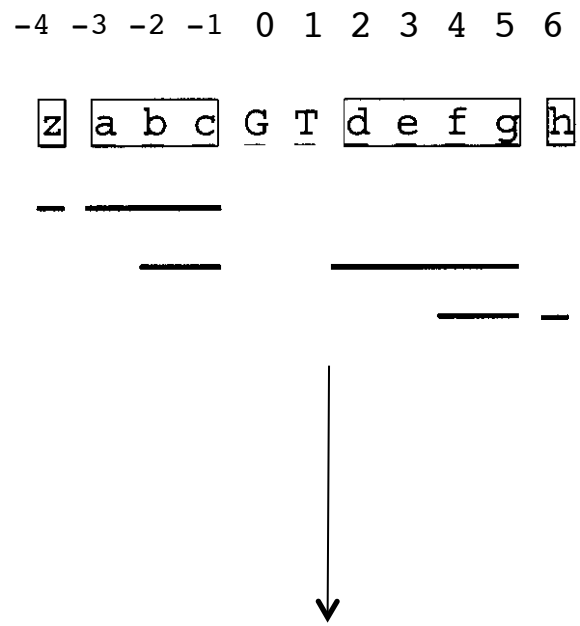


$$MI(X,Y) = \sum_{i=1}^{n}\sum_{j=1}^{n} P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i)P(y_j)}$$

Levine & Durbin 2001

Mutual information



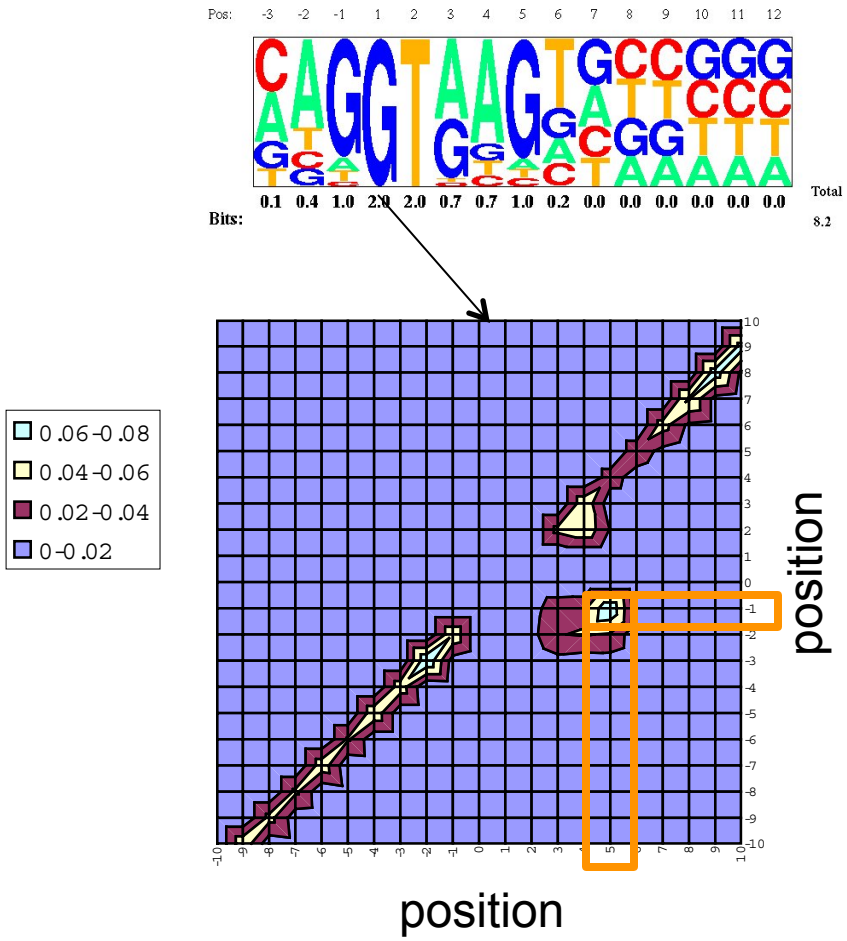


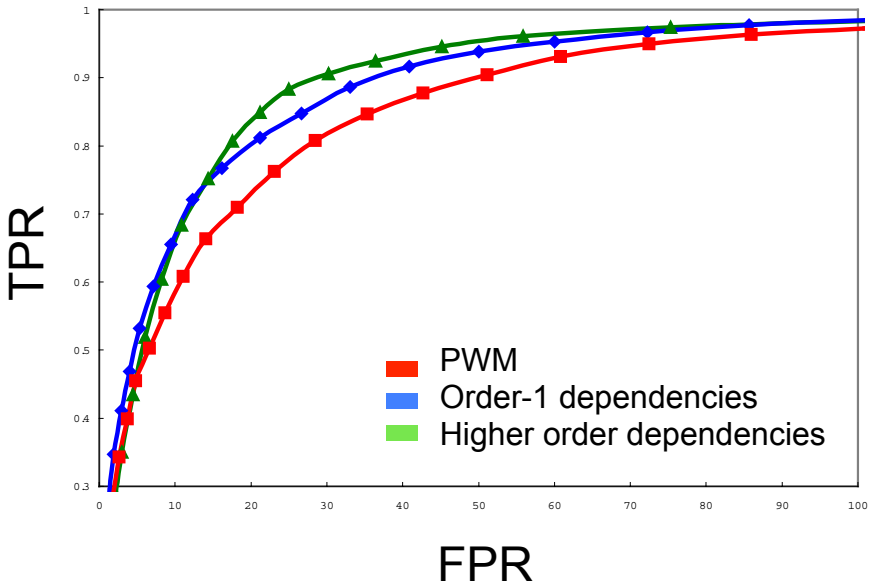
$$S(X) = \log_2 \frac{f(abc \mid z) * f(defg \mid bc) * f(h \mid fg)}{q(z)q(a \mid z)...q(c \mid b) * q(d)q(e \mid d)...q(h \mid g)}$$

The dependencies are used to build the model (see more later)

Levine & Durbin 2001

Mutual information





Using a model with dependencies improves the overall accuracy

Levine & Durbin 2001

Markov models

Mutual information can help us find out about dependencies between positions (between variables)

How to incorporate that into the model?

Markov models

The probability to observe a sequence according to the model described by *P*

$$S = s_1 s_2 s_3 \ldots s_N \qquad P(S) = P(s_1 s_2 s_3 \ldots s_N)$$

The joint probability can be re-written as a factorization of conditional probabilities

$$P(S) = P(s_1 s_2 s_3 \ldots s_N) = P(s_N \mid s_1 \ldots s_{N-1}) P(s_{N-1} \mid s_1 \ldots s_{N-2}) \cdot \ldots \cdot P(s_2 \mid s_1) P(s_1)$$

(the chain rule for probabilities)

For three elements:

$$P(s_1 s_2 s_3) = P(s_3 \mid s_1 s_2) P(s_1 s_2)$$

Apply twice the definition of conditional probability

$$P(s_3 \mid s_1 s_2) = \frac{P(s_1 s_2 s_3)}{P(s_1 s_2)} \qquad P(s_2 \mid s_1) = \frac{P(s_1 s_2)}{P(s_1)}$$

$$\boxed{P(s_1 s_2 s_3) = P(s_3 \mid s_1 s_2) P(s_2 \mid s_1) P(s_1)}$$

Markov models

The chain rule for probabilities

$$P(S) = P(s_1 s_2 s_3 \ldots s_N) = P(s_N \mid s_1 \ldots s_{N-1}) P(s_{N-1} \mid s_1 \ldots s_{N-2}) \cdot \ldots \cdot P(s_2 \mid s_1) P(s_1)$$

We define the **order** of the Markov chain as the number of the dependencies

ORDER 0	$P(s_i \mid s_1 \ldots s_{i-1}) = P(s_i)$
ORDER 1	$P(s_i \mid s_1 \ldots s_{i-1}) = P(s_i \mid s_{i-1})$
ORDER 2	$P(s_i \mid s_1 \ldots s_{i-1}) = P(s_i \mid s_{i-2} s_{i-1})$
…	…
ORDER n	$P(s_i \mid s_1 \ldots s_{i-1}) = P(s_i \mid s_{i-n} \ldots s_{i-1})$

E.g. Markov model of order 1 (Markov chain)

$$P(S) = P(s_1 s_2 s_3 \ldots s_N) = P(s_N \mid s_{N-1}) P(s_{N-1} \mid s_{N-2}) \cdot \ldots \cdot P(s_2 \mid s_1) P(s_1)$$

Markov models

E.g. Markov model of order 1:

$$P(S) = P(s_1 s_2 s_3 \ldots s_N) = P(s_N \mid s_{N-1}) P(s_{N-1} \mid s_{N-2}) \cdot \ldots \cdot P(s_2 \mid s_1) P(s_1)$$

1)Probabilities are estimated regardless of the position
(recall the NB model for book classification). E.g. for order 1

$$P(s_i = G \mid s_{i-1} = T) = \frac{n(s_i = G \mid s_{i-1} = T)}{\sum_{a \in M} n(s_i = a \mid s_{i-1} = T)}$$

2) We always need an "initial" set of probabilities. Eg. For order 1:

$$P(A) = \frac{n(A)}{\sum_{b \in M} n(b)}$$

These are estimated from the (**initial**) **positions** of the **training set**

Markov models

Example: consider the following sequences:

```
GCCGCGCTTG
GCTTGGTGGC
TGGCCGTTGC
```

Markov models

Example: consider the following sequences:

GCCGCGCTTG

GCTTGGTGGC

TGGCCGTTGC

$$P(C \mid G) = \frac{7}{12}$$

7 GC transitions

12 G positions

For the 1st order parameter, we count the number of times that *c* follows a g in the sequences

Markov models

Example: consider the following sequences:

GCCGCGCTTG
GCTTGGTGGC
TGGCCGTTGC

$$P(C \mid G) = \frac{7+1}{12+4} \qquad P(C) = \frac{0+1}{3+4}$$

$$P(A \mid G) = \frac{0+1}{12+4} \qquad P(A) = \frac{0+1}{3+4}$$

$$P(G \mid G) = \frac{3+1}{12+4} \qquad P(G) = \frac{2+1}{3+4}$$

$$P(T \mid G) = \frac{2+1}{12+4} \qquad P(T) = \frac{1+1}{3+4}$$

Initial probabilities

We can use pseudocounts with the sequences as before

Markov chains

Markov model of order 1 are generally called Markov chains

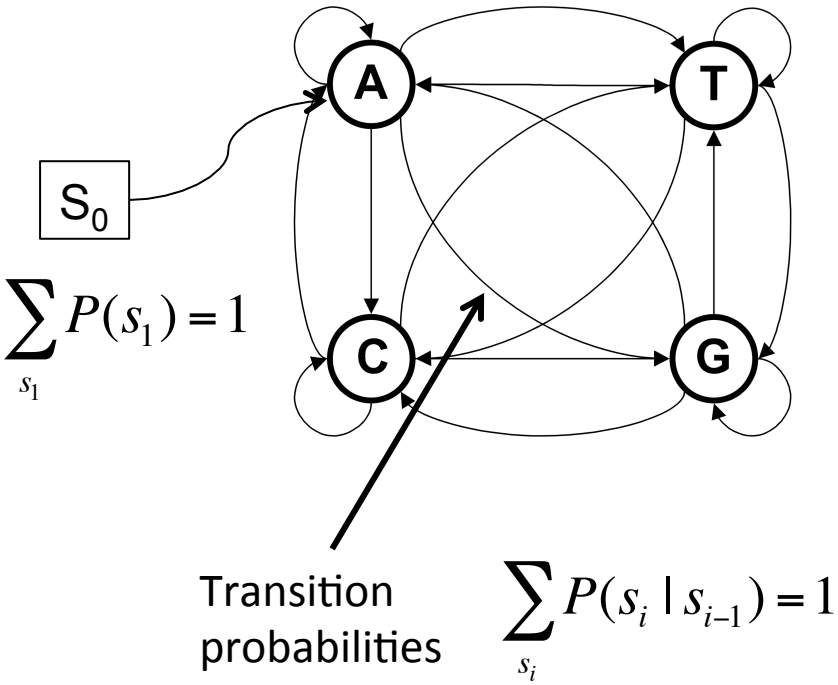
$$P(s_i \mid s_1 \ldots s_{i-1}) = P(s_i \mid s_{i-1})$$

$$P(S) = P(s_1 s_2 s_3 \ldots s_N) = P(s_N \mid s_{N-1}) P(s_{N-1} \mid s_{N-2}) \cdot \ldots \cdot P(s_2 \mid s_1) P(s_1)$$

E.g.: A Markov chain for nucleotides is a set of probabilities of the form *P(a|b)*, where *a,b = {A,C,G,T}*

We can view this as transitions…

Markov chains



A Markov chain can be represented as a set of states (1 per nucleotide)

with connections between them transition probabilities

The start of the sequence string is modeled with a "initial" fictitious state S_0

$$P(A \mid C) + P(C \mid C) + P(G \mid C) + P(T \mid C) = 1$$

Markov models

Markov model of order k: next base depends on previous k bases

For order 2:

$$P(S) = P(s_1 s_2 s_3 \ldots s_N)$$
$$= P(s_N \mid s_{N-2} s_{N-1}) P(s_{N-1} \mid s_{N-3} s_{N-2}) \cdot \ldots \cdot P(s_3 \mid s_1 s_2) P(s_1 s_2)$$

$$\mathrm{P}(ACA) = \mathrm{P}(A \mid AC)\mathrm{P}(AC)$$

Markov models

For order 2:

1) If we estimate these probabilities regardless of the position

$$P(s_i = G \mid s_{i-2} = A, s_{i-1} = T) = \frac{n(s_i = G \mid s_{i-2} = A, s_{i-1} = T)}{\sum_{a \in M} n(s_i = a \mid s_{i-2} = A, s_{i-1} = T)}$$

2) We need an "initial" set of probabilities: $P(s_1 s_2)$

$$P(AC) = \frac{n(AC)}{\sum_{a \in M} \sum_{b \in M} n(a,b)}$$

These are estimated from the "initial positions of the **training set**

Markov models

How to select the order?

The number of parameters (probabilities) to estimate grows exponentially with the order: $\sim 4^{(k+1)}$

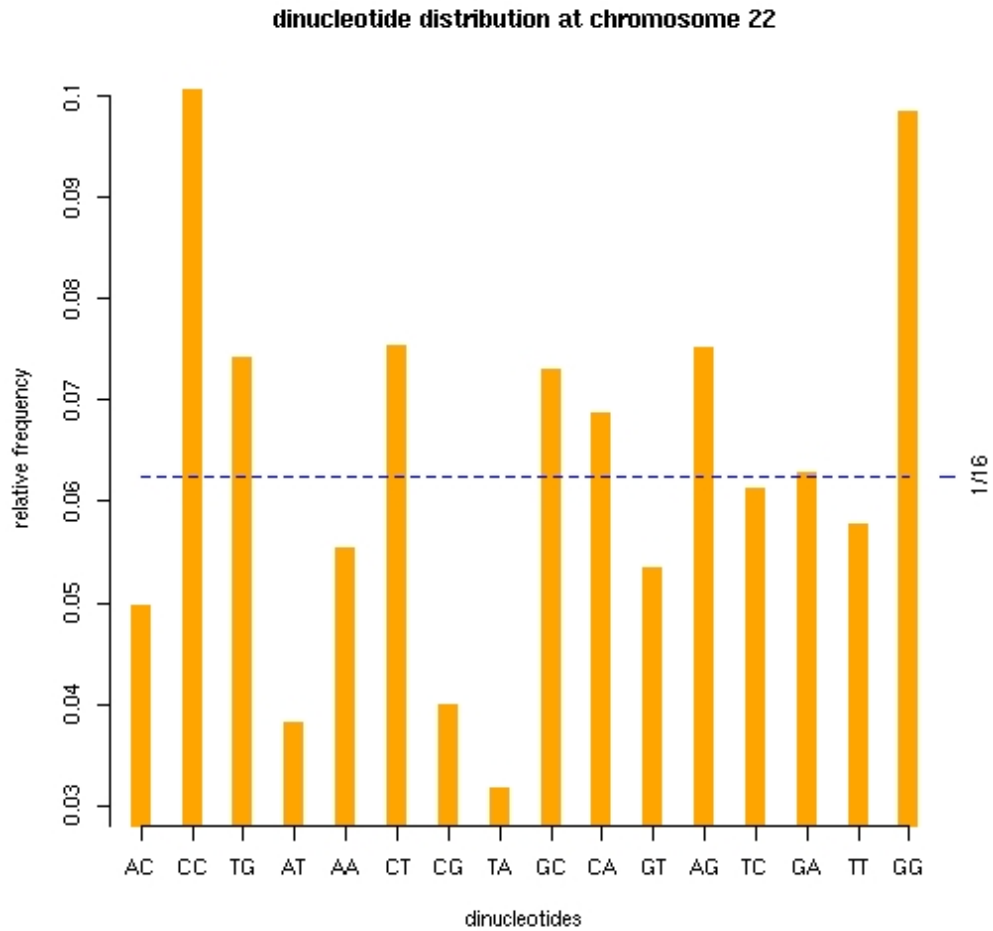
Higher order may be more accurate (captures better the dependencies)

But it is less reliable: less data to estimate parameters (more dependent on pseudocounts)

Example: CpG Islands

Example: CpG Islands

Wherever there is a CG (CpG) in the genome, C tends to change chemically by methylation. Methylated C is more likely to mutate into a T during replication. Thus, CpG dinucleotides are less frequent than expected:



Example: CpG Islands

This transformation is often suppressed in specific regions, like the promoter of some genes, giving rise to a high-content of CpGs: These regions are called CpG islands

CpG islands are of variable length, between hundreds to thousands of base pairs.

We would like to answer the following questions:

1) given a DNA sequence, Is it part of a CpG island?

2) Given a large DNA region, can we find CpG islands in it?

Example: CpG Islands

a^{+}_{st} Transition probability between two adjacent positions in CpG islands

a^{-}_{st} Transition probability between two adjacent pos. outside CpG islands

Given a sequence *S* the log-likelihood ratio is:

$$\sigma = \log LR = \log \frac{P(S\,|+)}{P(S\,|-)} = \sum_{i=1}^{N} \log \frac{a^{+}_{i-1,i}}{a^{-}_{i-1,i}}$$

(up to the contribution from the initial probabilities)

a^{+}_{st} a^{-}_{st}

+	A	C	G	T	-	A	C	G	T
A	0.180	0.274	0.426	0.120	A	0.300	0.205	0.285	0.210
C	0.171	0.368	0.274	0.188	C	0.322	0.298	0.078	0.302
G	0.161	0.339	0.375	0.125	G	0.248	0.246	0.298	0.208
T	0.079	0.355	0.384	0.182	T	0.177	0.239	0.292	0.292

The larger the value of sigma, the more likely is to be a CpG island

Example: CpG Islands

Approach 1: Given a large stretch of DNA of length any length

we calculate

$$\sigma = \log LR = \log \frac{P(S \mid +)}{P(S \mid -)} = \sum_{i=1}^{N} \log \frac{a^{+}_{i-1,i}}{a^{-}_{i-1,i}}$$

Sequences with $\sigma(S)$ > 0 are the possible CpG islands

Disadvantage: CpG islands may be much shorter than the whole sequence. We therefore could underscore the real CpG island by including too much false sequence.

As a result we will miss many positive cases.

Example: CpG Islands

Approach 2:
Given a large stretch of DNA of length *L*, we extract windows of *l* nucleotides:

$$S^{(k)} = (s_{k+1},\ldots,s_{k+l}) \qquad 1 \le k \le L - l \qquad l << L$$

For each window we calculate $\sigma(S^{(k)})$

$$\sigma(S^{(k)}) = \log LR = \log \frac{P(S\mid +)}{P(S\mid -)} = \sum_{i=1}^{k} \log \frac{a^{+}_{i-1,i}}{a^{-}_{i-1,i}}$$

Windows with $\sigma(S^{(k)}) > 0$ are the possible CpG islands

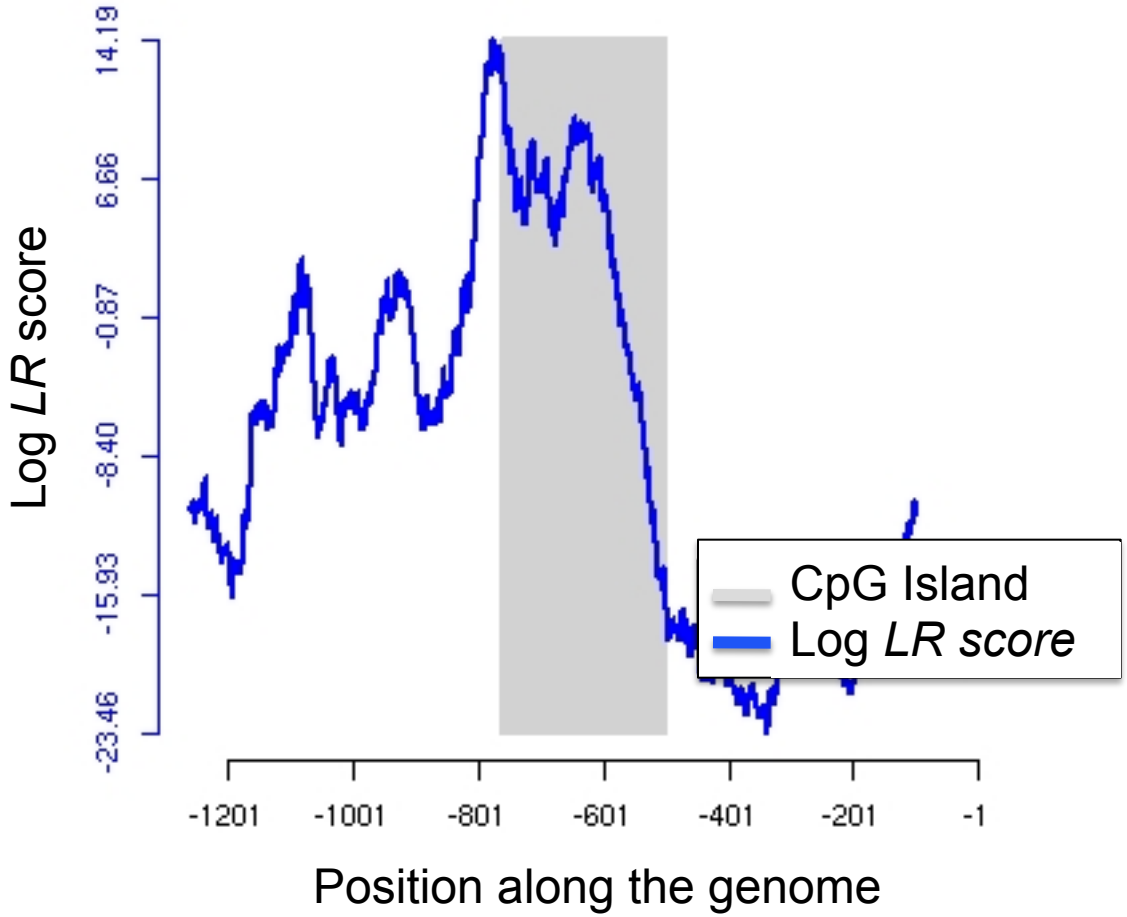
Disadvantage: We assume that CpG islands have at least *l* nucleotides. This must be fixed ad-hoc.

These Markov models do not provide a way of modeling the lengths.

Example: CpG Islands

For each window we calculate $\sigma(S^{(k)})$

$$\sigma(S^{(k)}) = \log L = \log \frac{P(S \mid +)}{P(S \mid -)} = \sum_{i=1}^{k} \log \frac{a^{+}_{i-1,i}}{a^{-}_{i-1,i}}$$



Exercise (from exam AGB 2014):

Consider the following sequence for a “C-island”:

`TCCCTCCCTCCC`

Estimate a Markov model of order 1 from this sequence.

Make a graphical representation of the model and calculate whether the sequence TCC belongs to the model. Assume that the background model is given by sequences with no frequency or positional preferences for T or C.

Help: you can use $\log_2 3 = 1.6$

Position-dependent Markov Models (Weight Array Matrices)

Inhomogeneous Markov models

We can model dependencies using conditional probabilities (Markov)

GGGGTGAGAGCATGCTGGTAAGCCCGGCTGGTG

$P(s_9 | s_8)$

Conditional probability

The Probability distribution is the same at every position

Inhomogeneous Markov models

E.g. Markov model of order n=2:

$$P(S) = P(s_1 s_2 s_3 \ldots s_N)$$
$$= P(s_N \mid s_{N-2} s_{N-1}) P(s_{N-1} \mid s_{N-3} s_{N-2}) \cdot \ldots \cdot P(s_3 \mid s_1 s_2) P(s_1 s_2)$$

Probabilities **may** correspond to the **different** distributions for **every position** and estimated from (n+1)-mer frequencies (3-mers in this case)

$$P(S) = P(s_1 s_2 s_3 \ldots s_N)$$
$$= P_N(s_N \mid s_{N-2} s_{N-1}) P_{N-1}(s_{N-1} \mid s_{N-3} s_{N-2}) \cdot \ldots \cdot P_3(s_3 \mid s_1 s_2) P_2(s_1 s_2)$$

Inhomogeneous Markov models

GGGGTGAGAGCATGCTGGTAAGCCCGGCTGGTG

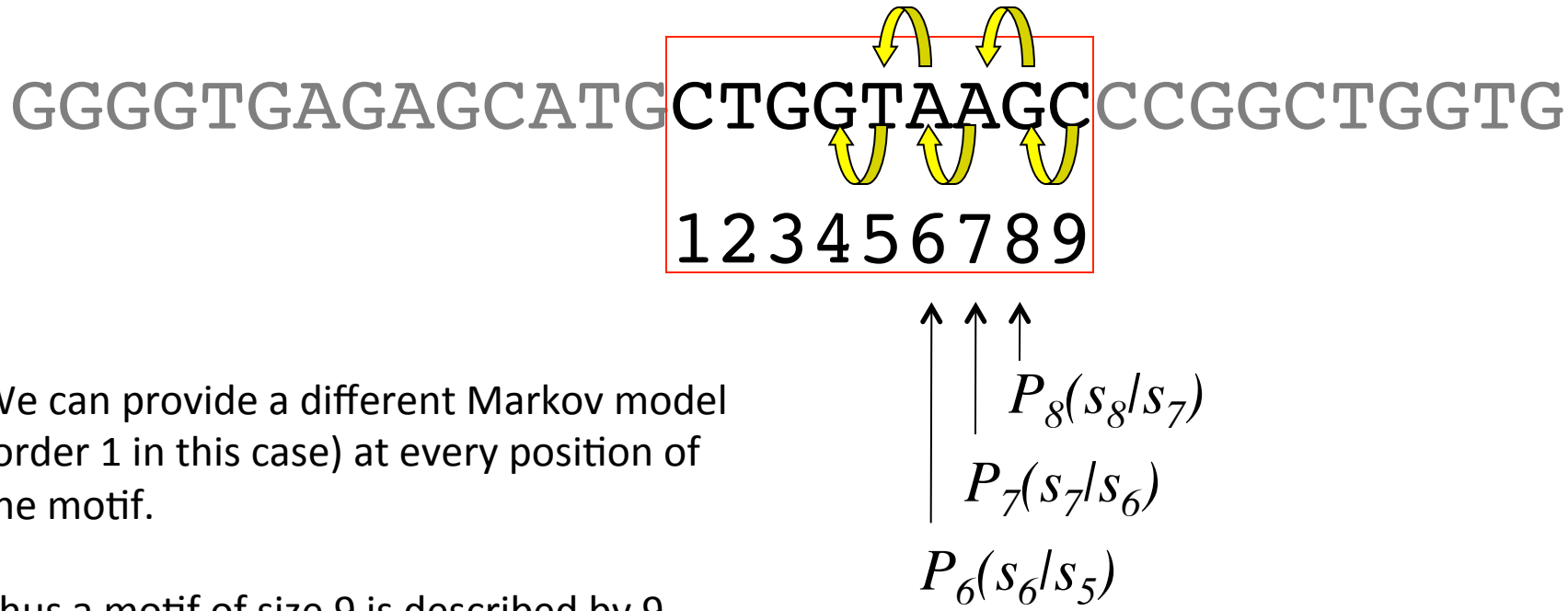
123456789

Position dependent model

Each position has a different probability distribution

Weight Array Matrices (WAMs) or Inhomogeneous Markov chains

Inhomogeneous Markov models



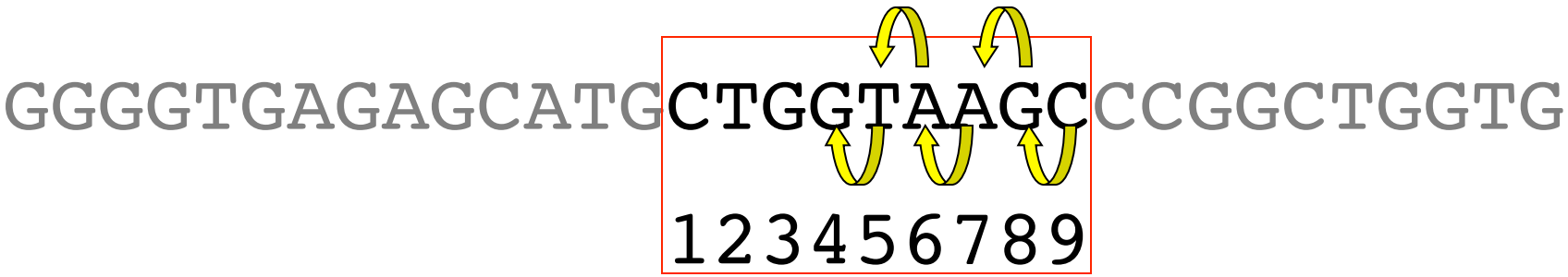
We can provide a different Markov model (order 1 in this case) at every position of the motif.

Thus a motif of size 9 is described by 9 Markov models:

$$P_9P_8P_7P_6P_5P_4P_3P_2P_1$$

One for each position

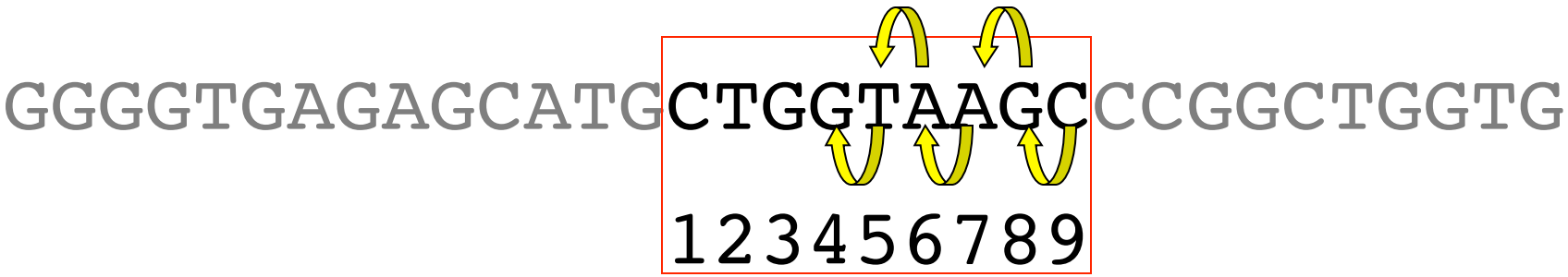
Inhomogeneous Markov models



$$P(S)=P_1(s_1)\ P_2(s_2|s_1)\ P_3(s_3|s_2)\ P_4(s_4|s_3)\ P_5(s_5|s_4)\ P_6(s_6|s_5)\ P_7(s_7|s_6)\ P_8(s_8|s_7)\ P_9(s_9|s_8)$$

One Markov model of order 0 (nucleotide frequency at a given position)

Inhomogeneous Markov models



$$P(S)=P_1(s_1)\ P_2(s_2|s_1)\ P_3(s_3|s_2)\ P_4(s_4|s_3)\ P_5(s_5|s_4)\ P_6(s_6|s_5)\ P_7(s_7|s_6)\ P_8(s_8|s_7)\ P_9(s_9|s_8)$$

8 Markov models of order 1 (transition matrices)

. . .	Position 2				Position 3 . . .				...
	A	C	G	T	A	C	G	T	
A	29.2	31.9	25.5	13.4	62.4	9.5	15.2	12.9	
C	48.6	32.5	6.2	12.7	69.2	11.6	6.4	12.8	
G	38.8	36.2	17.7	7.3	62.6	15.8	12.3	9.3	
T	16.4	41.3	29.5	12.9	17.7	25.6	29.5	27.2	

Inhomogeneous Markov models

We have used dependencies between adjacent positions.

But we can also model dependencies between any positions

Summary

Markov models allow to model dependencies in sequence data

Markov models are described by transition probabilities between states

Parameters are estimated from the observations by counting transitions

Order of the Markov models: higher order needs more data for training.
Generally we will use 1st order dependencies.

Homogeneous Markov models: non-positional dependence, e.g. CpG islands

Inhomogeneous Markov models: positional dependence, e.g. splice-sites

References

Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids
Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison.
Cambridge University Press, 1999

Problems and Solutions in Biological Sequence Analysis
Mark Borodovsky, Svetlana Ekisheva
Cambridge University Press, 2006

Bioinformatics and Molecular Evolution
Paul G. Higgs and Teresa Attwood.
Blackwell Publishing 2005.