Eukaryotic Gene Finding: The GENSCAN System

BMI/CS 776

www.biostat.wisc.edu/bmi776/

Spring 2012

Colin Dewey

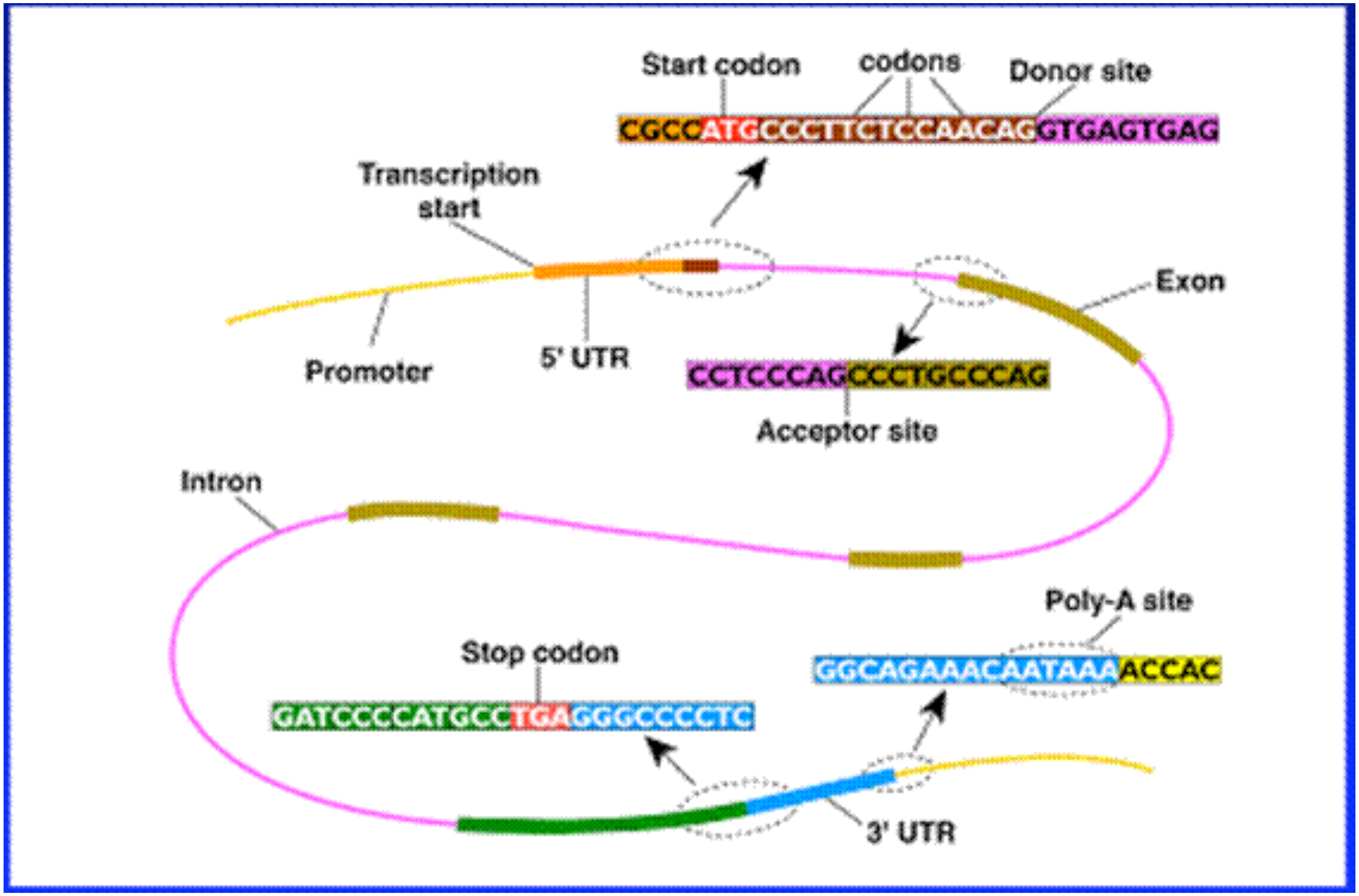
cdewey@biostat.wisc.edu

Goals for Lecture

the key concepts to understand are the following

- how knowledge about sequence elements can be used to make representational choices (topology, length distributions) in an HMM
- the MDD method
- understanding MDD as a graphical model

Eukaryotic Gene Structure



The GENSCAN HMM for Eukaryotic Gene Finding [Burge & Karlin '97]

Each shape represents a functional unit of a gene or genomic region

Pairs of intron/exon units represent the different ways an intron can interrupt a coding sequence (after 1st base in codon, after 2nd base or after 3rd base)

Complementary submodel (not shown) detects genes on opposite DNA strand

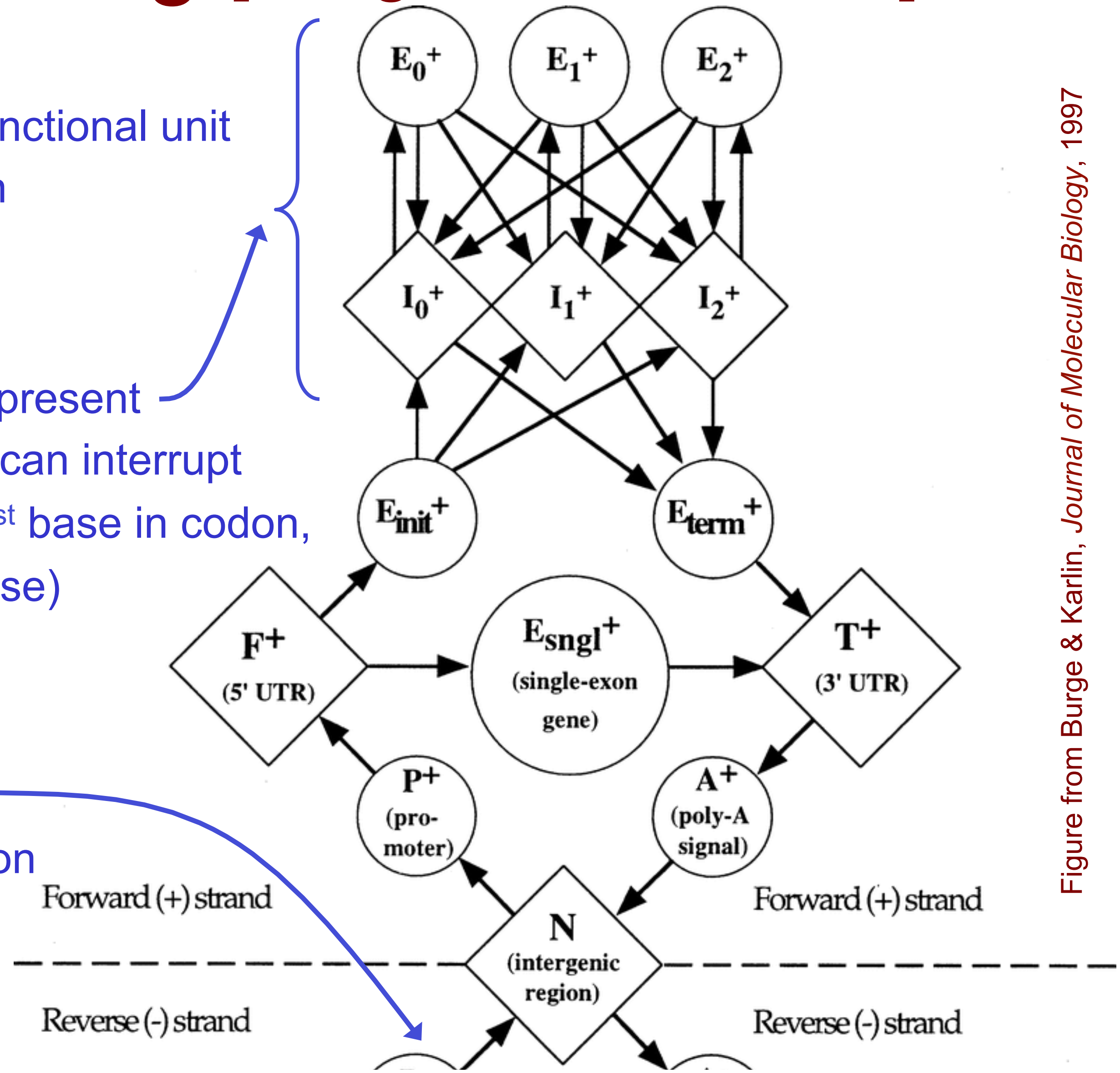


Figure from Burge & Karlin, *Journal of Molecular Biology*, 1997

The GENSCAN HMM

- for each sequence type, GENSCAN models
  - the length distribution
  - the sequence composition

- length distribution models vary depending on sequence type
  - * nonparametric (using histograms)
  - parametric (using geometric distributions)
  - fixed-length

- sequence composition models vary depending on type
  - 5th-order, inhomogeneous
  - 5th -order homogenous
  - 1st-order inhomogeneous
  - * tree-structured variable memory (MDD)

The GENSCAN HMM

- semi-Markov models are well motivated for some sequence elements (e.g. exons)
- dependency structure of splice sites motivates the use of MDD models, which can represent context-specific dependencies

Length Distributions of Introns/Exons

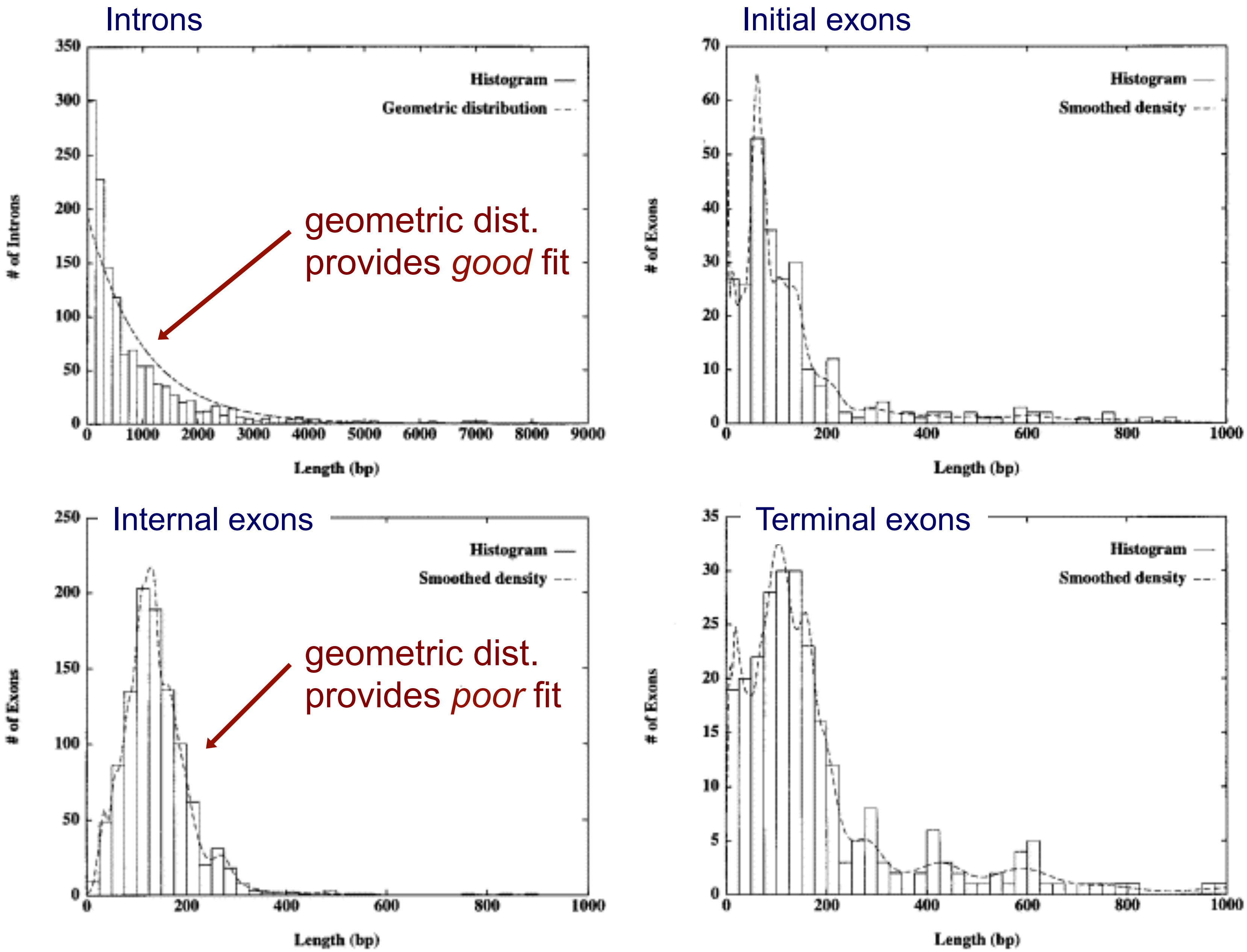
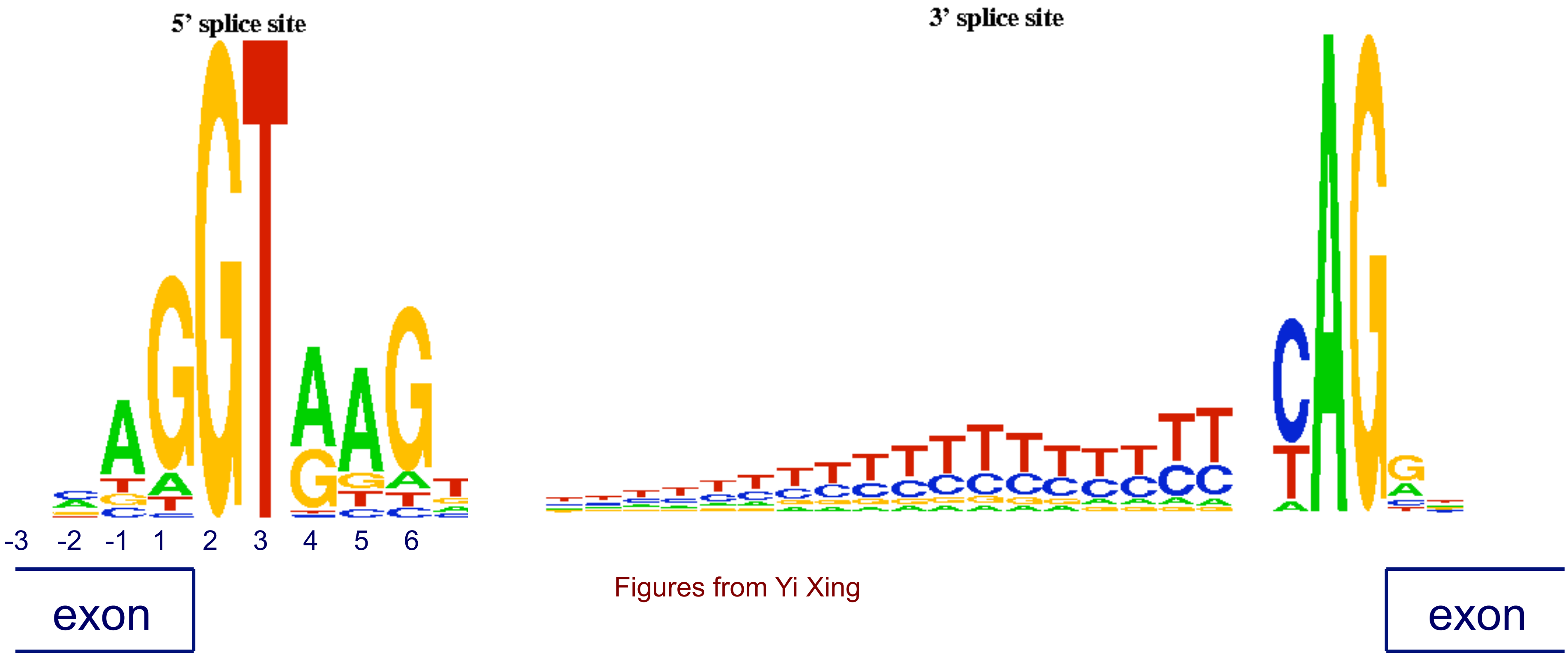


Figure from Burge & Karlin, *Journal of Molecular Biology*, 1997

Splice Signals

donor sites | *acceptor* sites



Figures from Yi Xing

- there are significant dependencies among non-adjacent positions in donor splice signals

Motivation for MDD

- How can we detect significant dependencies between non-adjacent positions?

```
ATGGGTCCATCTACATATACACATCCATT
TATCTCTACCGCGCTAGCCTAGTCGGATT
GCTACGACCGCTAACAGCTCGACCTGTGA
CCTTCGGGCTATATATTATTCTTCTTATA
TCGAAATAGACTAGCTAAATCGCTAGCTA
TCCGCGCTCGCTAACAGCTACCAAATAGA
CGTAGCTAGATCGAATCGAAAGCCCTACT
ACACCAGGCTTCTAATCGATTAGATCCCA
```

i j

pos i matches consensus	**pos i does NOT match consensus**	
		pos j = A
		pos j = C
		pos j = G
		pos j = T

- compute χ^2 values using 2×4 table
  - **alternative hypothesis**: distribution for column j depends on whether the consensus base is in column i
  - **null hypothesis**: distribution for column j is the same in both cases

Motivation for MDD

- table shows χ^2 values for pairs of positions around donor sites
- values marked with * show statistically significant dependency

Table 4. Dependence between positions in human donor splice sites: χ^2-statistic for consensus indicator variable C_i *versus* nucleotide indicator X_j

i	Con	j: −3	−2	−1	+3	+4	+5	+6	Sum
−3	c/a	—	61.8*	14.9	5.8	20.2*	11.2	18.0*	131.8*
−2	A	115.6*	—	40.5*	20.3*	57.5*	59.7*	42.9*	336.5*
−1	G	15.4	82.8*	—	13.0	61.5*	41.4*	96.6*	310.8*
+3	a/g	8.6	17.5*	13.1	—	19.3*	1.8	0.1	60.5*
+4	A	21.8*	56.0*	62.1*	64.1*	—	56.8*	0.2	260.9*
+5	G	11.6	60.1*	41.9*	93.6*	146.6*	—	33.6*	387.3*
+6	t	22.2*	40.7*	103.8*	26.5*	17.8*	32.6*	—	243.6*

The Maximal Dependence Decomposition (MDD) Approach

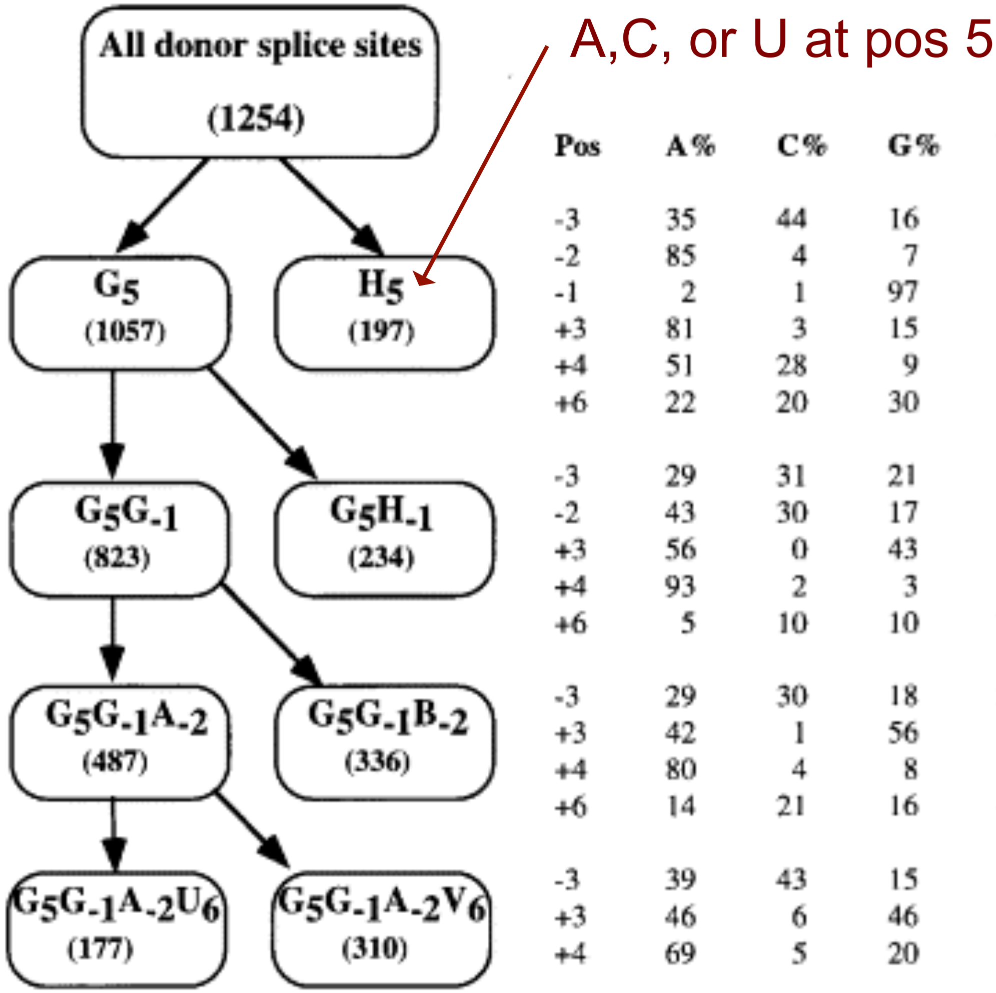
- induce a tree that represents the dependency structure apparent in the data

- induce partial position weight matrices for each node and leaf of tree

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

- use the tree + weight matrices to calculate the probability of a given sequence

The Structure of An MDD Learned Tree



Pos	A%	C%	G%	U%
-3	33	36	19	13
-2	56	15	15	15
-1	9	4	78	9
+3	44	3	51	3
+4	75	4	13	9
+6	14	18	19	49

-3	34	37	18	11
-2	59	10	15	16
+3	40	4	53	3
+4	70	4	16	10
+6	17	21	21	42

-3	37	42	18	3
+3	39	5	51	5
+4	62	5	22	11
+6	19	20	25	36

-3	32	40	23	5
+3	27	4	59	10
+4	51	5	25	19

Pos	A%	C%	G%	U%
-3	35	44	16	6
-2	85	4	7	5
-1	2	1	97	0
+3	81	3	15	2
+4	51	28	9	12
+6	22	20	30	28

-3	29	31	21	18
-2	43	30	17	11
+3	56	0	43	0
+4	93	2	3	3
+6	5	10	10	76

-3	29	30	18	23
+3	42	1	56	1
+4	80	4	8	8
+6	14	21	16	49

-3	39	43	15	2
+3	46	6	46	3
+4	69	5	20	7

Figure from Burge & Karlin, *Journal of Molecular Biology*, 1997

Explaining a Sequence with an MDD Tree

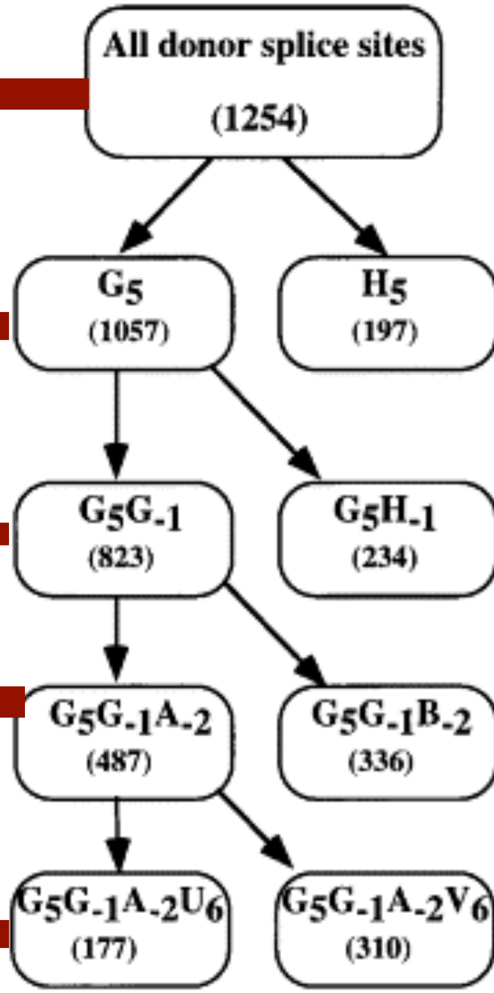
- shown are selected position weight matrices for the tree

	5
A	0.1
C	0.1
G	0.7
U	0.1

	-1
A	0.1
C	0.1
G	0.7
U	0.1

	-2
A	0.5
C	0.3
G	0.1
U	0.1

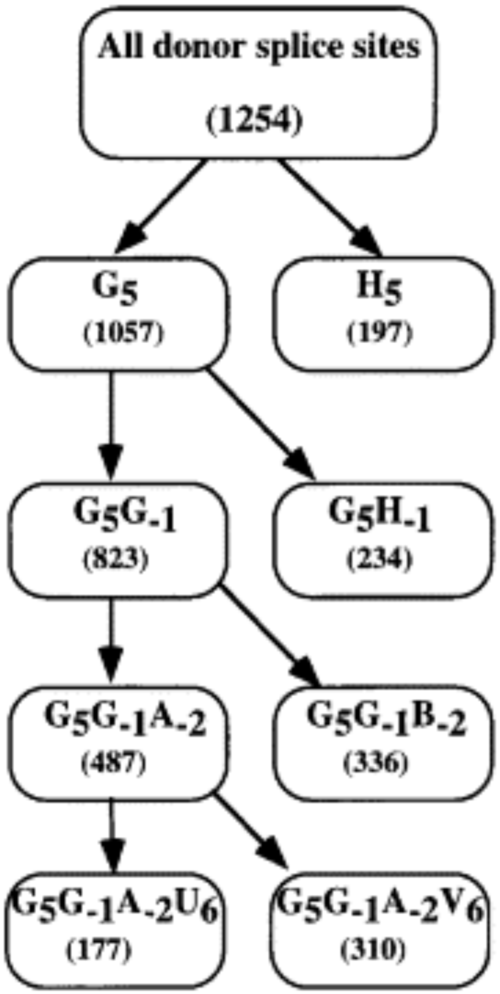
	6
A	0.1
C	0.2
G	0.1
U	0.6



	-3	-2	-1	3	4	5	6
A	0.3	0.4		0.2	0.5		0.1
C	0.4	0.3		0.1	0.1		0.1
G	0.2	0.2		0.6	0.2		0.1
U	0.1	0.1		0.1	0.2		0.7

	-3	-2	-1	3	4	5	6
A	0.3			0.2	0.5		
C	0.4			0.1	0.1		
G	0.2			0.6	0.2		
U	0.1			0.1	0.2		

Explaining a Sequence with an MDD Tree



calculate P(x_5)

if $x_5 \neq G$, use the weight matrix for H_5 subset

else

calculate P(x_{-1}) from from G_5 subset

if $x_{-1} \neq G$, use the WM for G_5H_{-1} subset

else

calculate Pr(x_{-2}) from G_5G_{-1} subset

⋮

Explaining a Sequence with an MDD Tree

- using model from previous slide

$$P(\text{AAGGUCAGU}) = 0.3 \times 0.5 \times 0.7 \times 1 \times 1 \times 0.1 \times 0.5 \times 0.7 \times 0.6$$

-3 -1 1 6

The MDD Algorithm: Finding the Tree

Given: a set of aligned training sequences T
positions $P = \{1, \ldots, k\}$
tree = find_MDD_subtree(T, P)

find_MDD_subtree(T, P)
for each position i in P
    determine the consensus base C_i
    calculate dependence between C_i, other positions
if stopping criteria not met
    choose the value of i such that S_i is maximal
    make a node with C_i as the test
    create a single-column PWM for position i
    D_i^+ = sequences in T with base C_i at position i
    D_i^- = other sequences
    left subtree = find_MDD_subtree(D_i^+, $P - \{ i \}$)
    right subtree = find_MDD_subtree(D_i^-, $P - \{ i \}$)
else
    create a partial PWM for remaining positions in P

test for position j conditioned on match to consensus at i

$$S_i = \sum_{j \neq i} \chi^2(C_i, x_j)$$

Stopping Criteria for MDD

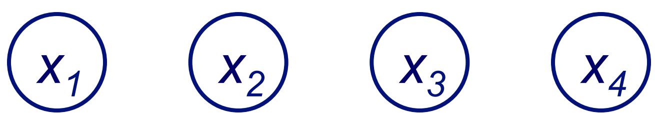
1. the $(k\text{-}1)^{th}$ level is reached; no further positions to split on

2. no significant dependencies between positions are detected

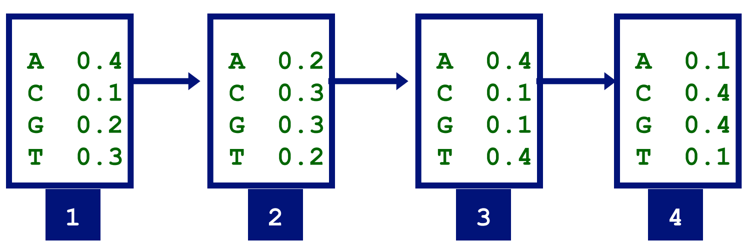
3. number of sequences in given subset is sufficiently small

A Graphical View of Dependency Structure

- we can represent the dependency structure of a sequence model as a graph
  - nodes represent sequence positions
  - edges represent dependencies in probability distribution
- the dependency structure of a 0th order Markov chain of length 4 (e.g. a motif model inferred by MEME) :

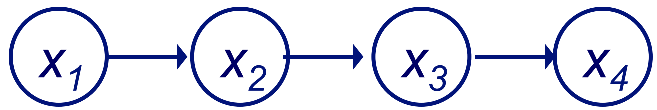


- note: this is different than the transition graph

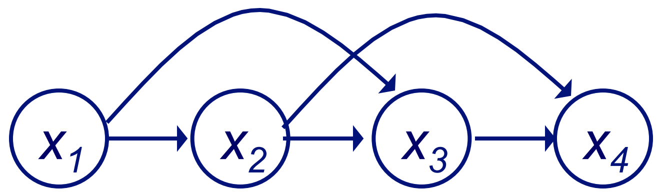


A Graphical View of Dependency Structure

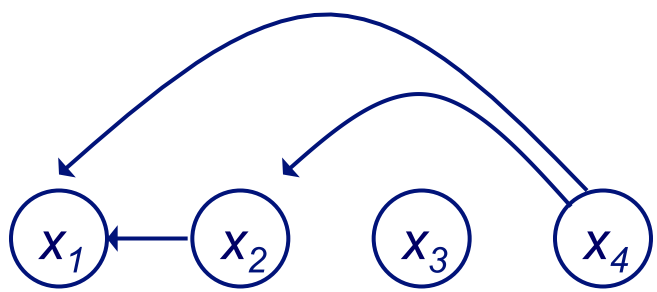
- 1^{st} order model



- 2^{nd} order model

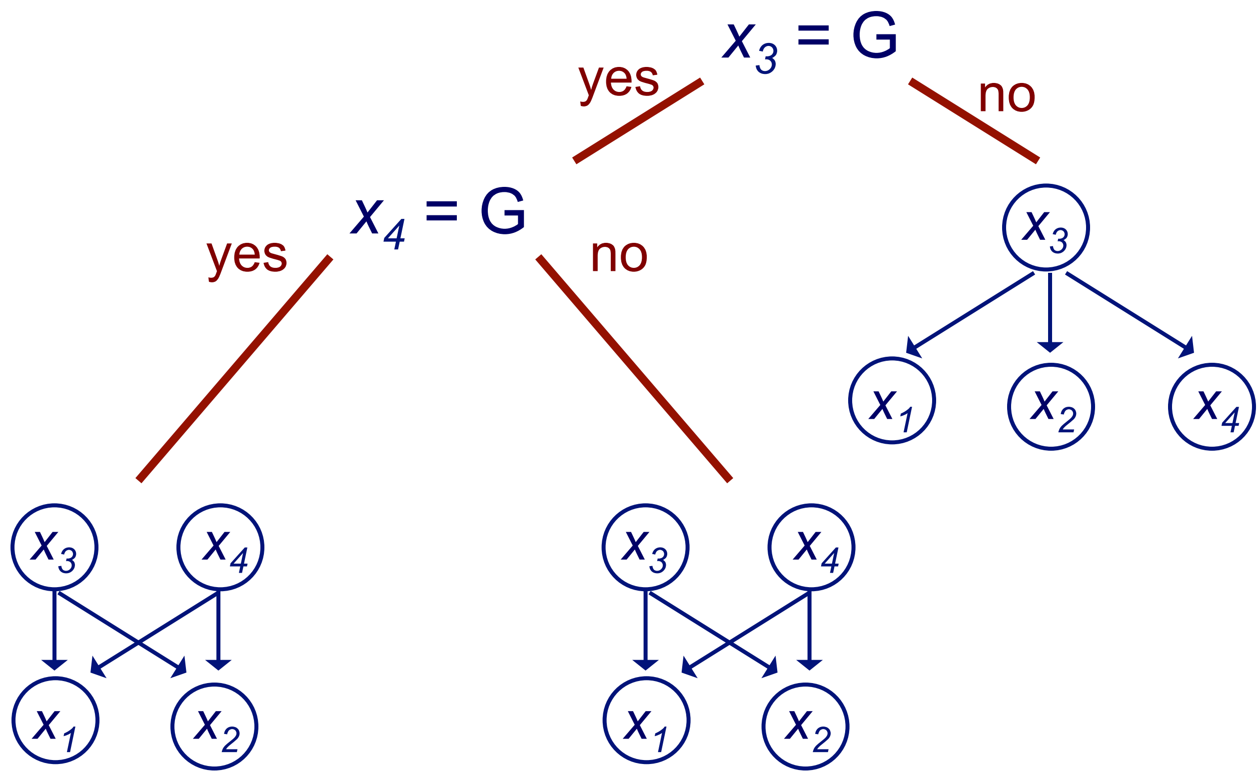


- for a fixed-length model, we could consider arbitrary dependencies



A Graphical View of Dependency Structure

- MDD allows arbitrary dependencies conditioned on *values* of certain variables



GENSCAN Conclusions

- HMMs readily enable background knowledge to be incorporated into the model
  - state topology
  - length distributions
  - order of Markov chains

- key technical ideas
  - semi-Markov models (previously developed): can represent arbitrary length distributions
  - MDD: can represent context-specific dependencies