MIT OpenCourseWare
http://ocw.mit.edu

6.047 / 6.878 Computational Biology: Genomes, Networks, Evolution
Fall 2008

For information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms.

Computational Gene Prediction and Generalized Hidden Markov Models

Today

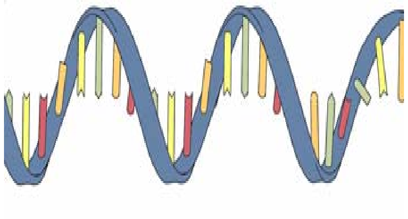
- Gene Prediction Overview
- HMMs for Gene Prediction
- GHMMs for Gene Prediction
- Genscan

Genome Annotation

Genome Sequence

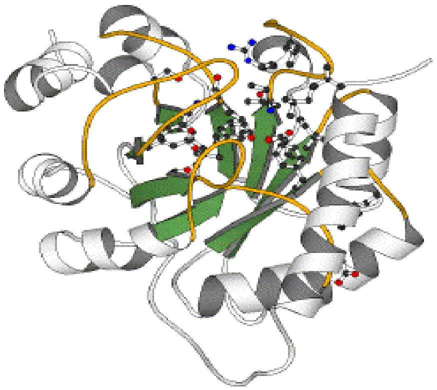
```
> HSCKIIBE, Human gene for casein kinase II subunit beta (EC 2.7.1.37).
ggggctgagatgtaaattagaggagctggagaggagtgcttcagagtttgggttgctttaagaaagggt
ggttccgaattctcccgtggttggagggccgaatgtgggaggagggaggataccagaggcagggaagga
gaacttgagctttactgacactgttctttttctagctgacgtgaagatgagcagctcagaggaggtgtc
ctggatttcctggttctgtgggctccgtggcaatgaattcttctgtgaagtgagttctcttcaacctcc
ctacttgccagcttcacatatcttcccaccagacgttccttcacatattccacttctacactgttctct
aaagcttttatgggagagagtgtaggtgaactagggagagacacaagtacttctgctgagttgggagtg
agaaacaagcacaacagatgcagttgtgttgatgataaggcatcacttagagcattttgcccaggtcaa
agatgaggattttgatatgggttccctcttggcttccatgtcctgacaggtggatgaagactacatcca
ggacaaatttaatcttactggactcaatgagcaggtccctcactatcgacaagctctagacatgatctt
ggacctggagcctggtgaggcaccctcagggttgttttgtgtgtgtgcgtgcactatttttctcttcaa
atctctattcacttgcctgaattttgccaaatttcctttggttctctgatttctttaaccccaaattca
tgctttattttgatcctccacctgactcttgtctagttttgtgacgtatatcacttgttctcatgtttt
tgcaagggtcagaagcccaggtttctgggtcccatgcccagatgttggatggggtaaggcccaaaagta
ggtgctaggcaaactgaatagcccgcagcccctggatatgggcagggcacctaggaaagctgaaaaaca
agtagttgcatttggccgggctgtggttcagatgaagaactggaagacaaccccaaccagagtgacctg
attgagcaggcagccgagatgctttatggattgatccacgcccgctacatccttaccaaccgtggcatc
gcccagatggtgaggcctctctgctcctacctgcctccttctgagcagtaagagacacaggttcctgca
gcaagaagtcatgtttaagccctgtttaaggaagctagctgagaagaggggaagaaccccagaacttgg
ccctgccctaatttggaagaaaggcaacacagaagtttgagagcccatctagtccagagaagggggcct
ctggacagagttggaaggagtgccgacagagttggtatgggttgggctgcgaagggagttgcctcttct
ttacatctacctgccaaccccttccattgtattcacctcagttggaaaagtaccagcaaggagactttg
gttactgtcctcgtgtgtactgtgagaaccagccaatgcttcccattGgtgagtgttgaagaagggaaa
ggaaagcaccgtgtggcagtcttatgggaaggagttggggctcaacacattggagcctgagtcctgagg
ggaggttaggtaggaatagggggatacctggcctgctgagtctggctgtctcccaggcctttcagacat
cccaggtgaagccatggtgaagctctactgccccaagtgcatggatgtgtacacacccaagtcatcaag
acaccatcacacggatggcgcctacttcggcactggtttccctcacatgctcttcatggtgcatcccga
gtaccggcccaagagacctgccaaccagtttgtgcccaggtagggagcagggagagtcattaagggtca
aaggaaaggcccaagatcccccagagaggggaggacagggcatggccctttcttgaggtctgcttctcc
cagaatcagggcatctccctgctgagtgactgtgggaaagttatttgattatctgtgcttgagttacct
tattgtagaatgttcttgagctgagaagttgggaaccacgaggctttagctctgagcaggtccatagag
gagctcaggtggggaggtgggaatgcaggtgactggcagggcctggatggggctcatgctgctgcctct
ctgacctctgccctggcctaggctctacggtttcaagatccatccgatggcctaccagctgcagctcca
agccgccagcaacttcaagagcccagtcaagacgattcgctgattccctcccccacctgtcctgcagtc
tttgtcttttcctttctttttttgccacccctttcaggaaccctgtatggttttttagtttaaattaaagga
gtcgttatcgtggtgggaatatgaaataaagtagaagaaaaggccatgagctagtctgctggtgcttgc
ggaagggggtggagcgtggccatggaaatcgggctccacggcccagggatgg
```

Transcription



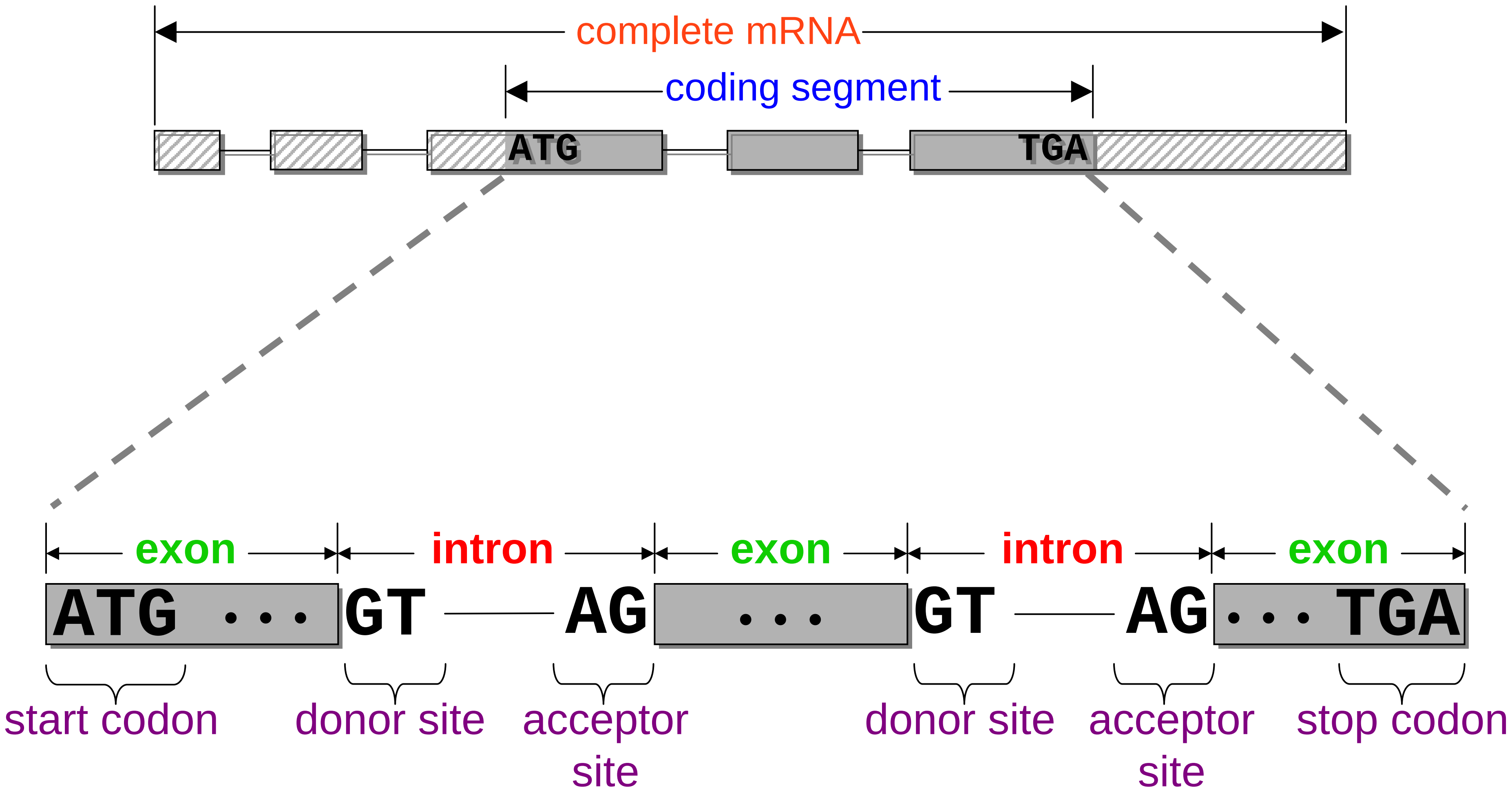
RNA

Translation



Protein

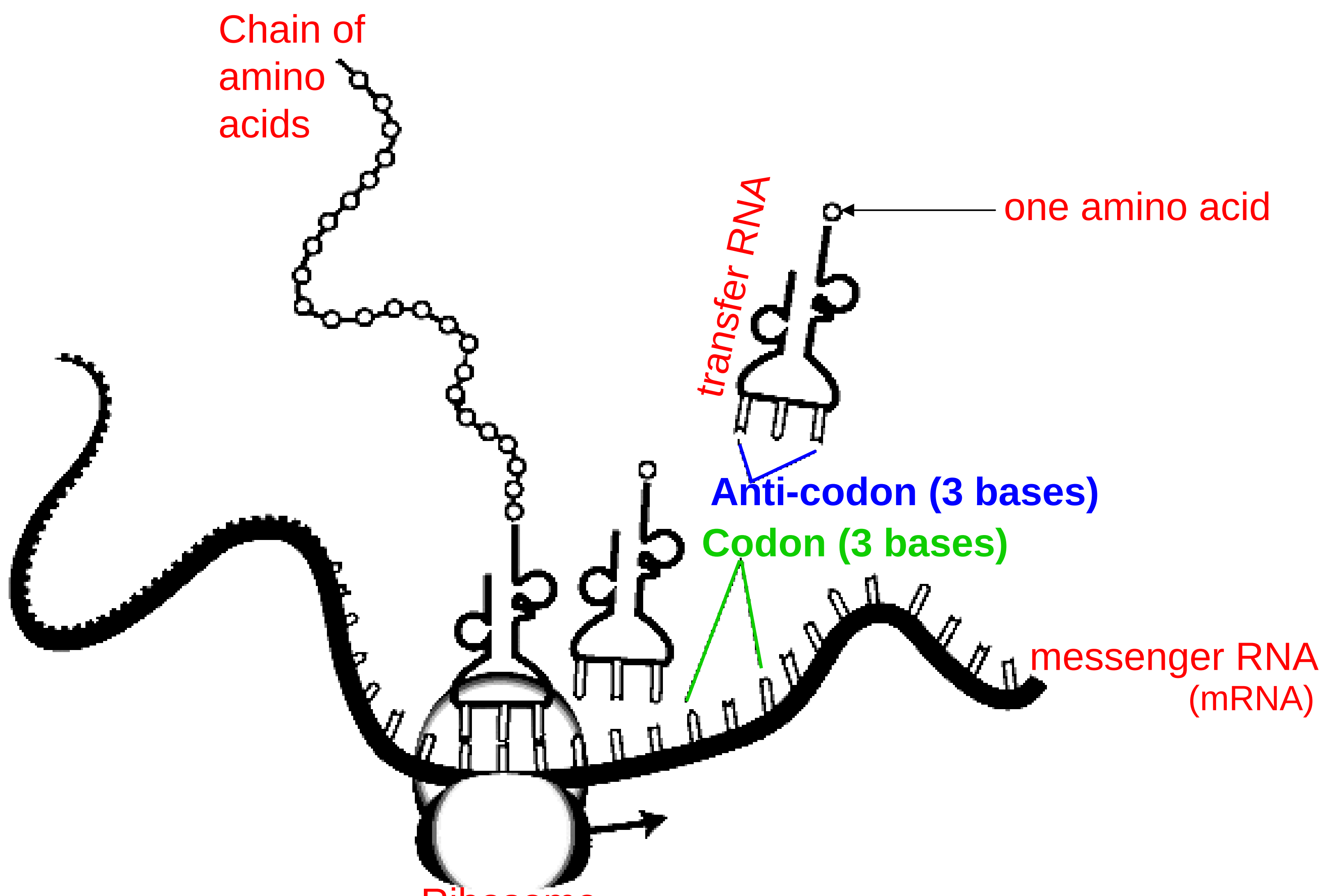
Eukaryotic Gene Structure



Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

Translation



Courtesy of William Majoros.Used with permission.

http://geneprediction.org/book/classroom.html

Genetic Code

Acid	Codons	Acid	Codons	Acid	Codons	Acid	Codons
A	GCA GCC GCG GCT	G	GGA GGC GGG GGT	M	ATG	S	AGC AGT TCA TCC TCG TCT
C	TGC TGT	H	CAC CAT	N	AAC AAT	T	ACA ACC ACG ACT
D	GAC GAT	I	ATA ATC ATT	P	CCA CCC CCG CCT	V	GTA GTC GTG GTT
E	GAA GAG	K	AAA AAG	Q	CAA CAG	W	TGG
F	TTC TTT	L	CTA CTC CTG CTT TTA	R	AGA AGG CGA CGC CGG CGT	Y	TAC TAT

Figure by MIT OpenCourseWare.

Each amino acid is encoded by one or more *codons*.

Each codon encodes a single *amino acid*.

The *third position* of the codon is the most likely to vary, for a given amino acid.

Gene Prediction as "Parsing"

- Given a genome sequence, we wish to label each nucleotide according to the parts of genes
  - Exon, intron, intergenic, etc
- The sequence of labels must follow the syntax of genes
  - e.g. exons must be followed by introns or intergenic not by other exons
- We wish to find the optimal parsing of a sequence by some measure

Features

A *feature* is *any DNA subsequence of biological significance*.

For practical reasons, we recognize two broad classes of features:

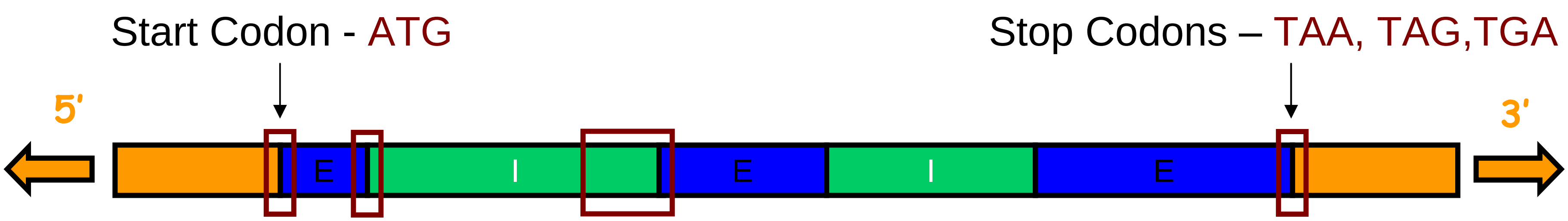
signals — short, fixed-length features

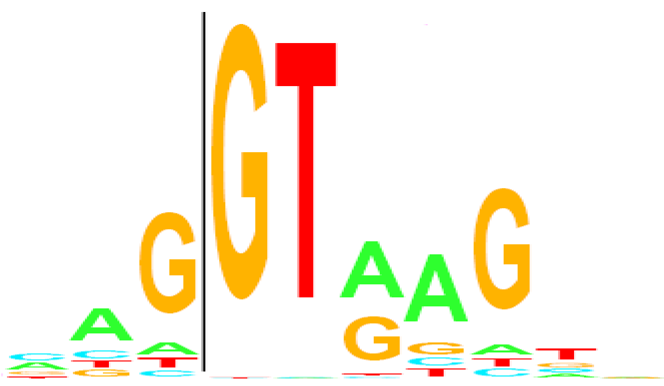
content regions — variable-length features

Courtesy of William Majoros. Used with permission.

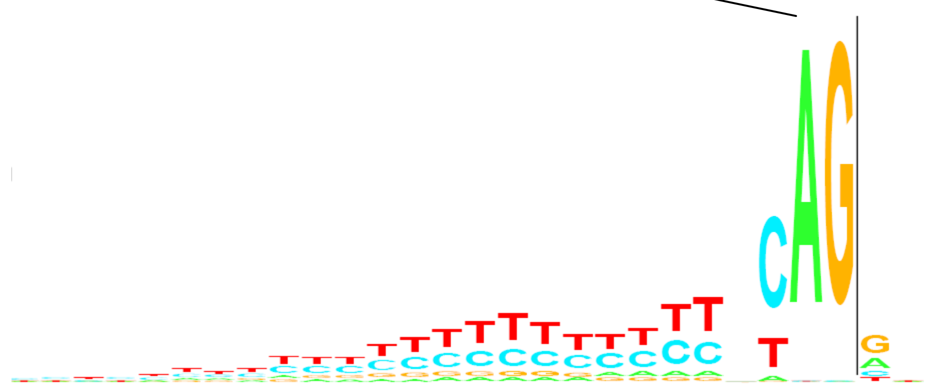
Signals

Associated with short fixed(-ish) length sequences





5' Splice Site (Acceptor)



3' Splice Site (Donor)

Content Regions

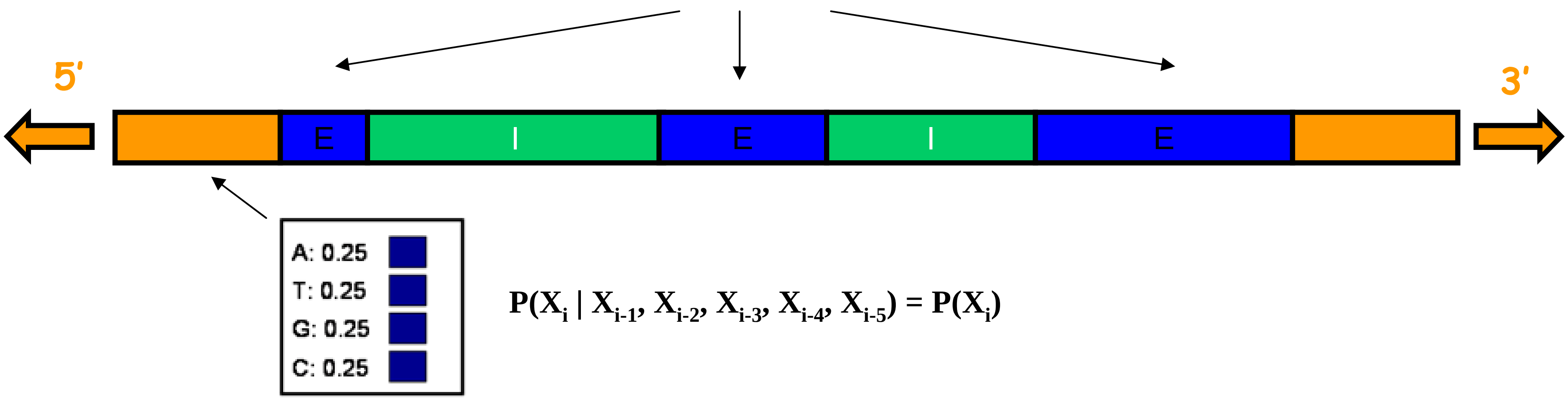
Content regions often have characteristic base composition

Example

- Recall: often multiple codons for each amino acid
- All codons are not used equally

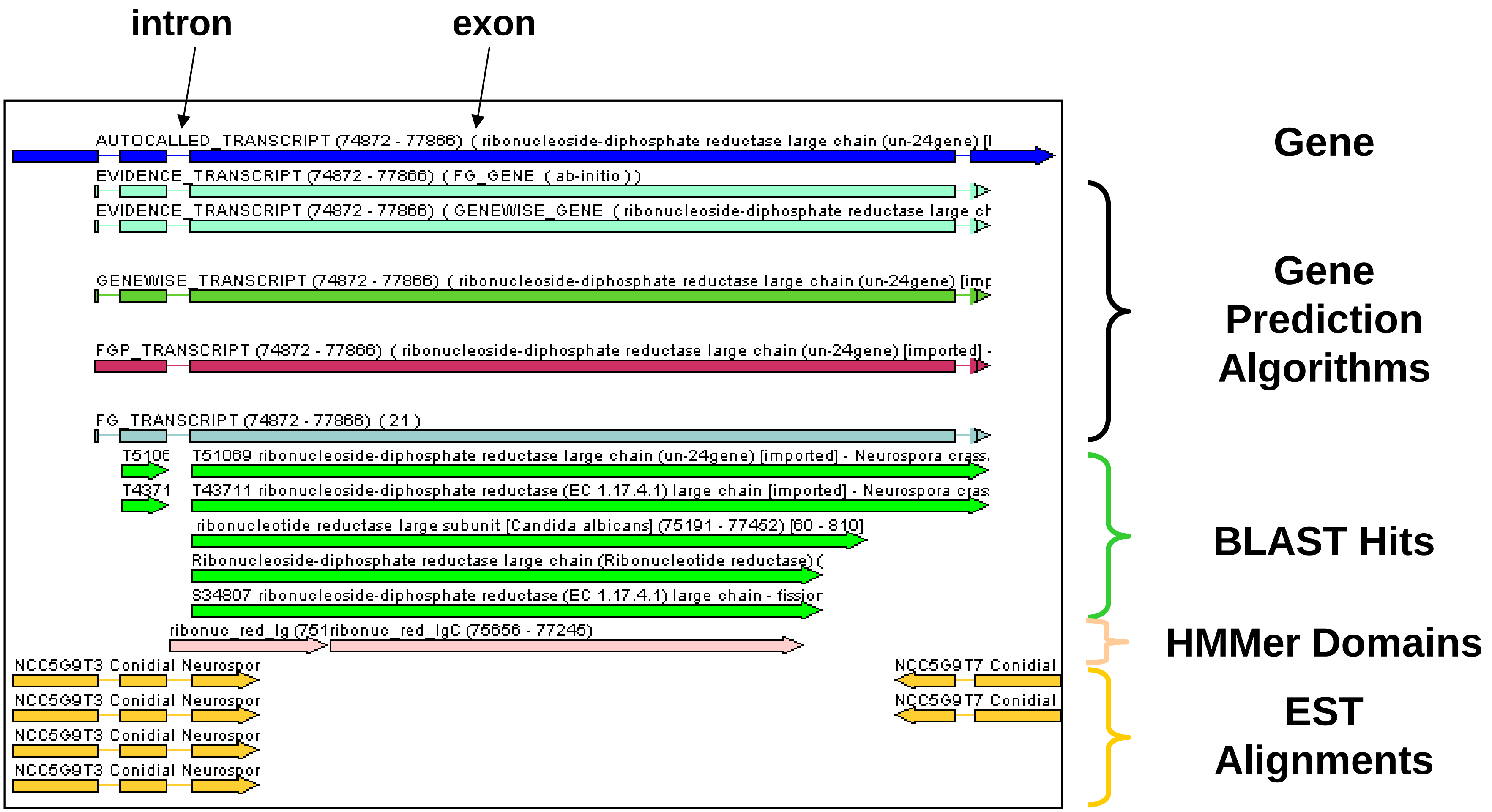
Characteristic higher order nucleotide statistics in coding sequences (hexanucleotides)

$$P_{exon}(X_i \mid X_{i-1}, X_{i-2}, X_{i-3}, X_{i-4}, X_{i-5})$$



$$P(X_i \mid X_{i-1}, X_{i-2}, X_{i-3}, X_{i-4}, X_{i-5}) = P(X_i)$$

Extrinsic Evidence



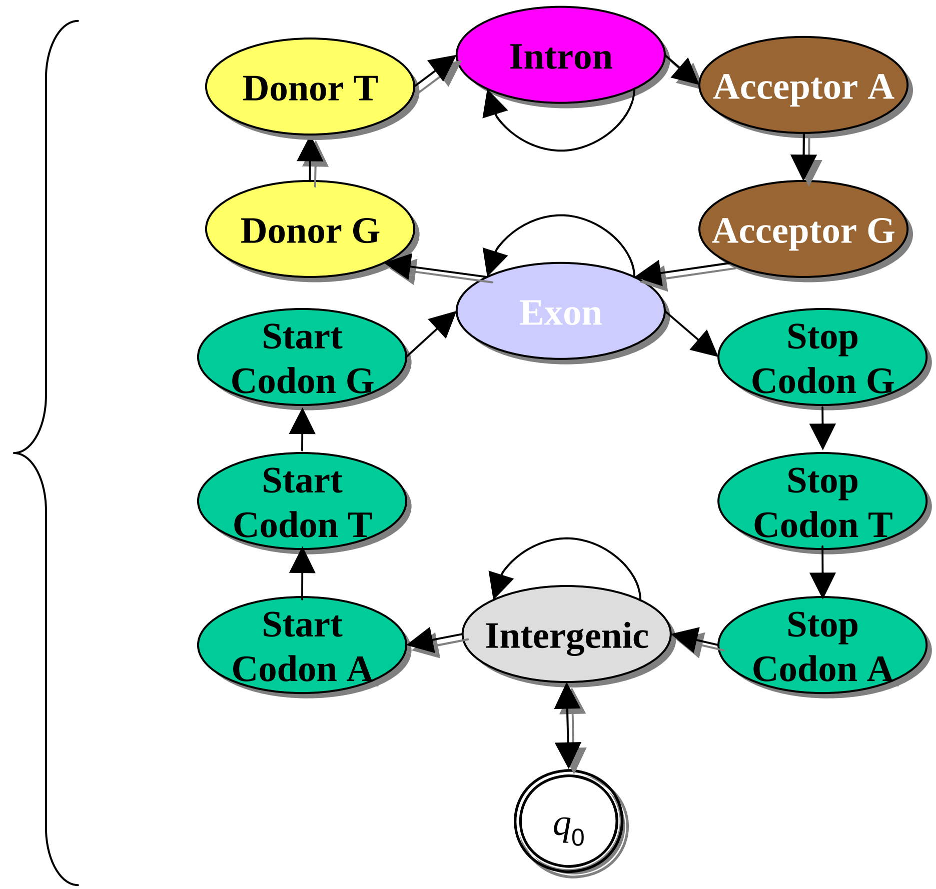
Neurospora crassa (a fungus)

HMMs for Gene Prediction

- States correspond to gene and genomic regions (exons, introns, intergenic, etc)
- State transitions ensure legal parses
- Emission matrices describe nucleotide statistics for each state

A (Very) Simple HMM

the Markov model:



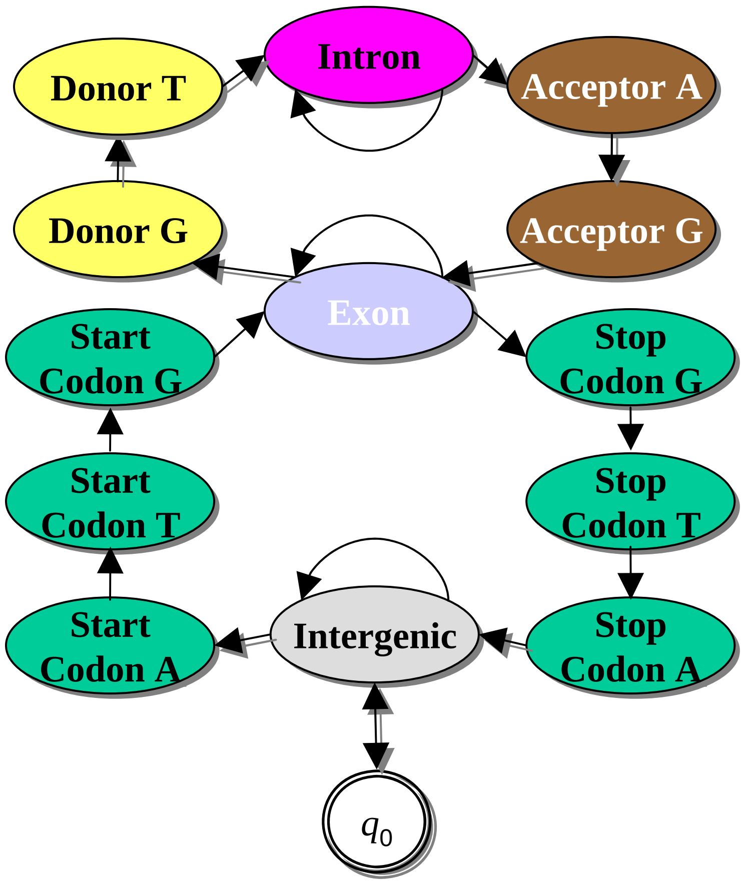
Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

A Generative Model

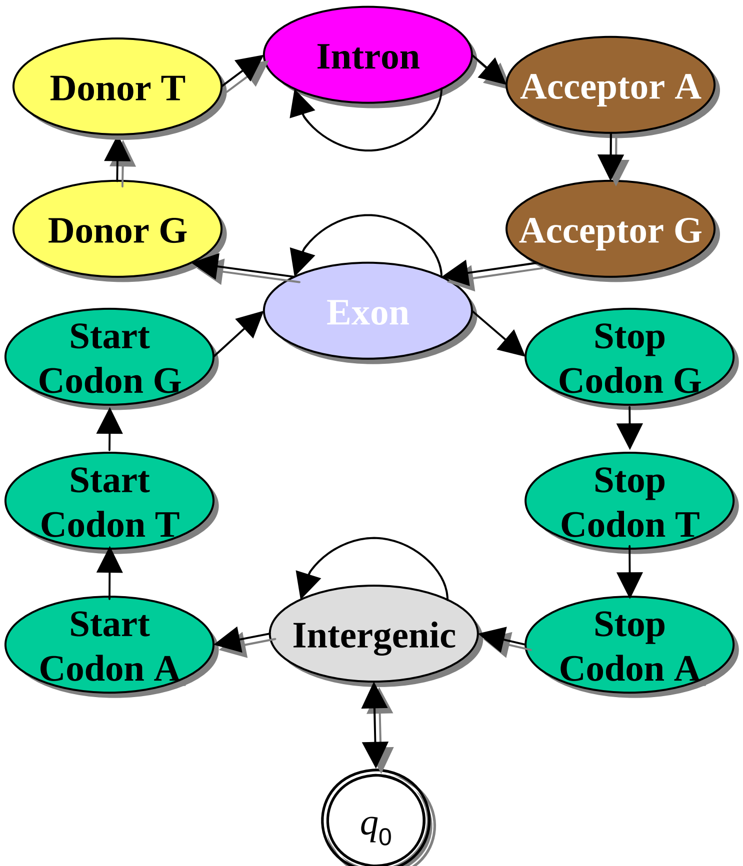
We can use this HMM to generate a sequence <u>and</u> state labeling

- The initial state is q_0
- Choose a subsequent state, conditioned on the current state, according to $a_{jk}=P(q_k|q_j)$
- Choose a nucleotide to emit from the state emissions matrix $e_k(X_i)$
- Repeat until number of nucleotides equals desired length of sequence



But We Usually Have the Sequence

the Markov model:



the input sequence: AGCTAGCAGTATGTCATGGCATGTTCGGAGGTAGTACGTAGAGGTAGCTAGTATAGGTCGATAGTACGCGA

What is the best state labeling?

Courtesy of William Majoros. Used with permission. http://geneprediction.org/book/classroom.html

Finding The Most Likely Path

- A sensible choice is to choose π that maximizes $P[\pi|x]$
- This is equivalent to finding path π^* that maximizes total joint probability $P[\,x, \pi\,]$:

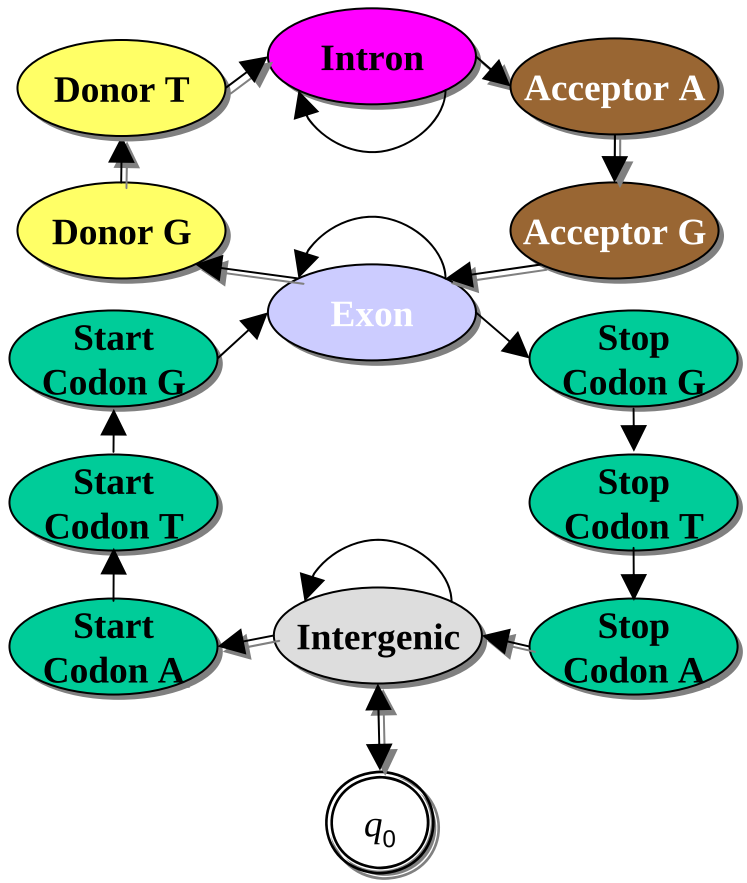
$$P(x,\pi) = a_{0\pi_1} * \prod_i e_{\pi_i}(x_i) \times a_{\pi_i \pi_{i+1}}$$

start emission transition

How do we select $\pi*$ efficiently?

A (Very) Simple HMM

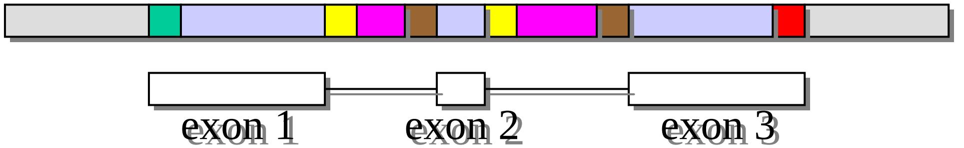
the Markov model:



the input sequence: AGCTAGCAGTATGTCATGGCATGTTCGGAGGTAGTACGTAGAGGTAGCTAGTATAGGTCGATAGTACGCGA

the most probable path:

the gene prediction:



Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

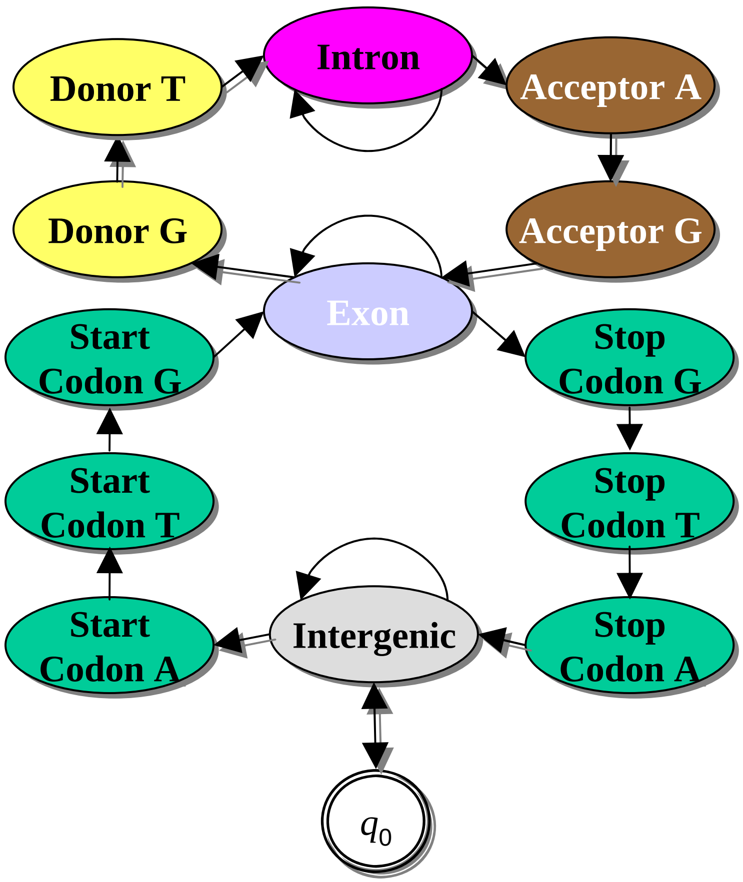
A Real HMM Gene Predictor

Title page of journal article removed due to copyright restrictions.
The article is the following:

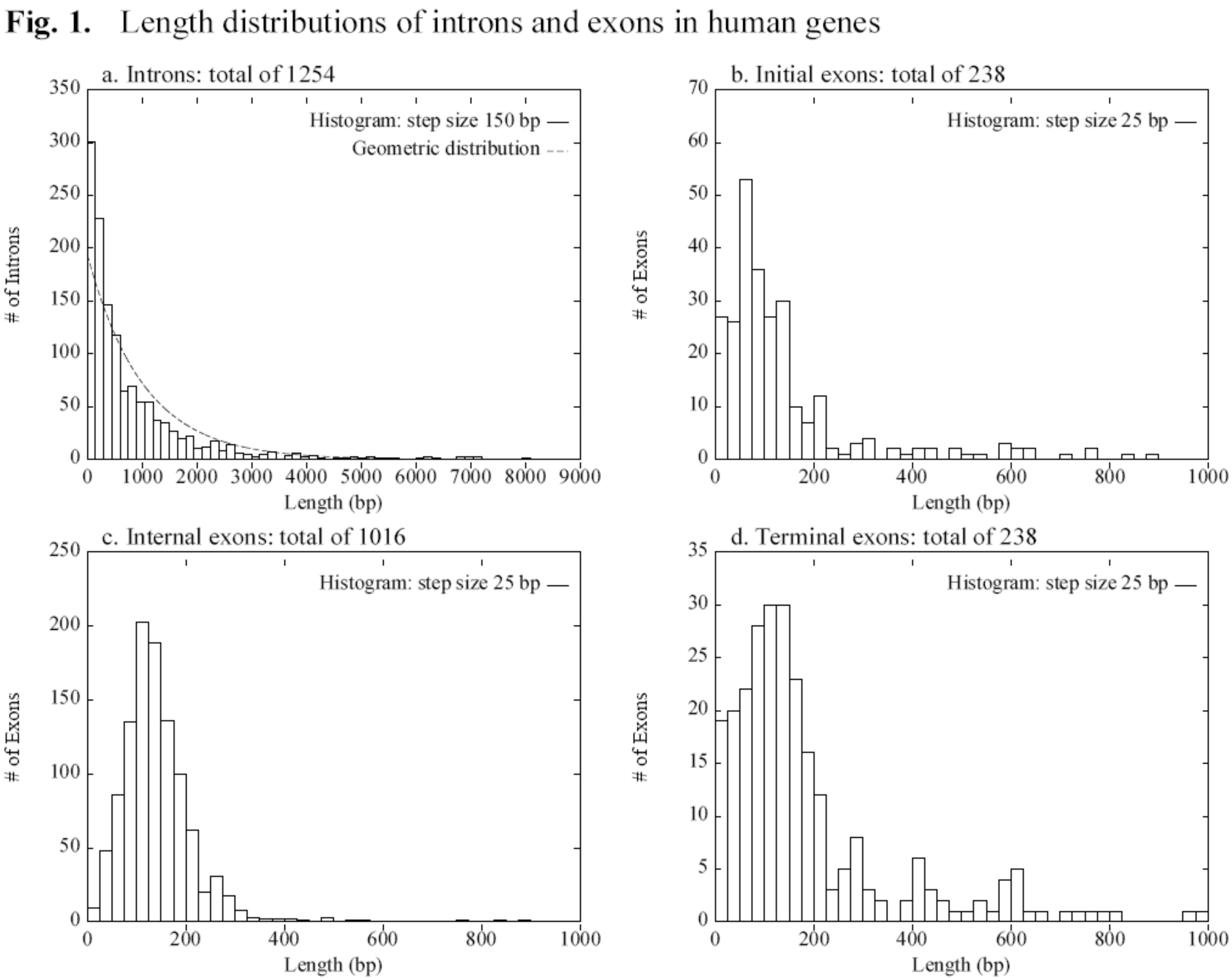
Krogh, Anders, I. Saira Mian, and David Haussler. "A Hidden Markov Model That Finds Genes in *E.coli* DNA." Nucleic Acids Research 22, no. 22 (1994): 4768-4778.

HMM Limitations

The HMM framework imposes constraints on state paths…



Human Exon Lengths



Legend. Intron, exon length data from 238 multi-exon genes of GENSCAN learning set (Appendix A).

Burge, MIT PhD Thesis

Courtesy of Christopher Burge. Used with permission.

Fungal Intron Lengths

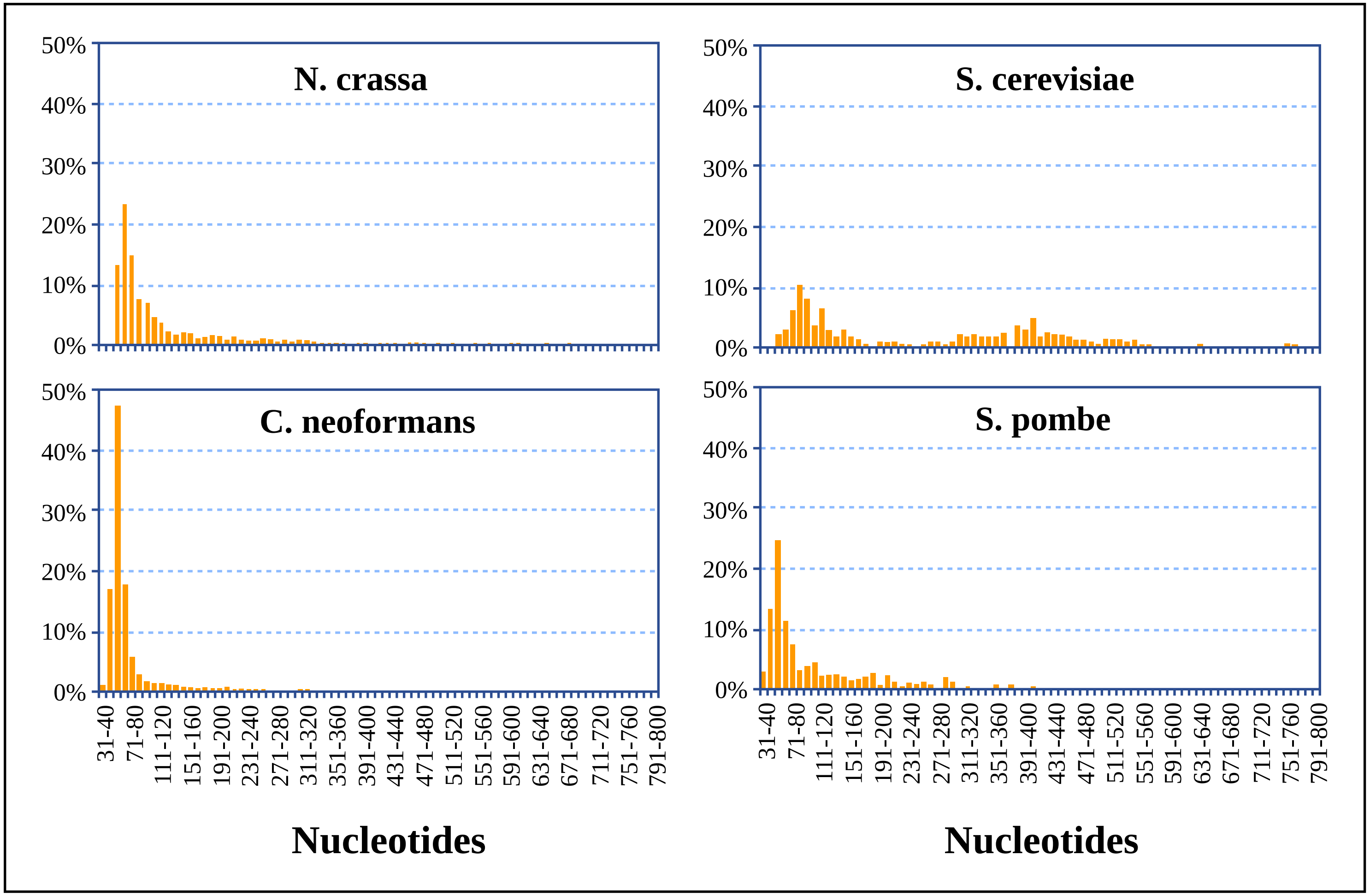
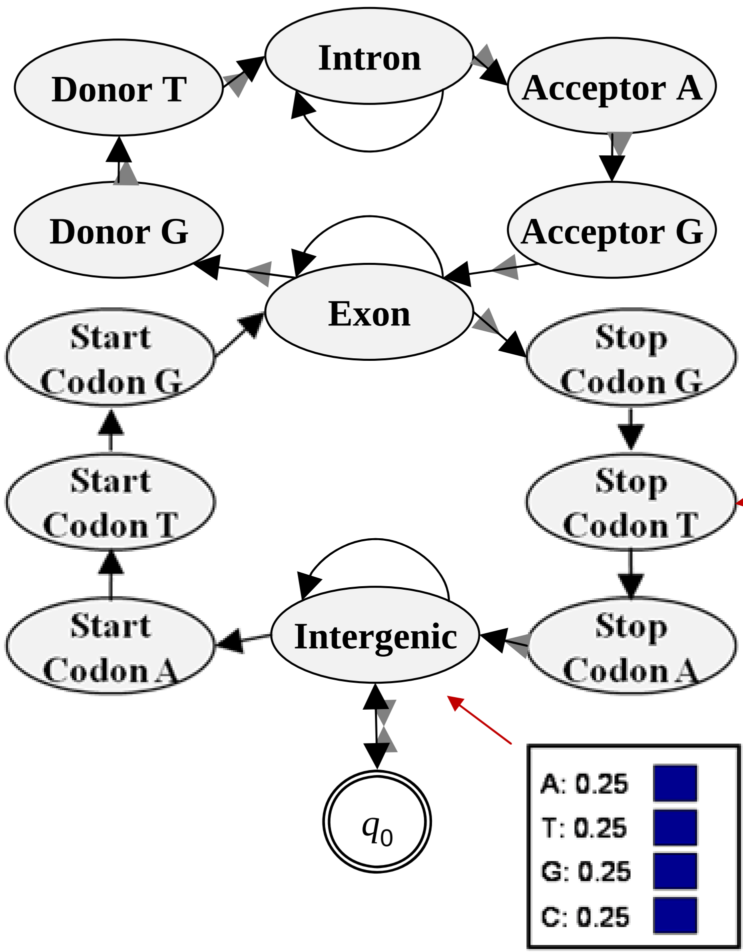


Figure by MIT OpenCourseWare.

Kupfer et al., (2004) Eukaryotic Cell

HMM Emissions

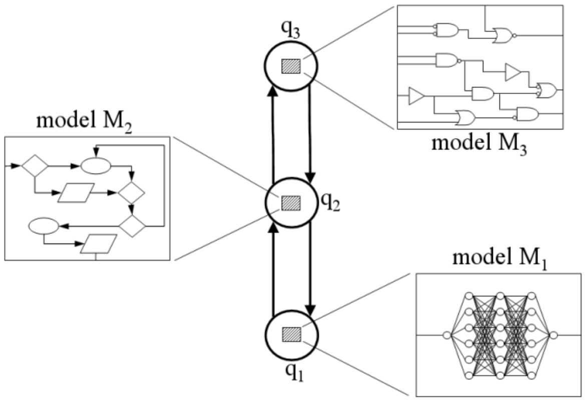
HMMs typically emit a single nucleotide per state



P(T)=1
P(A),P(G),P(C)=0

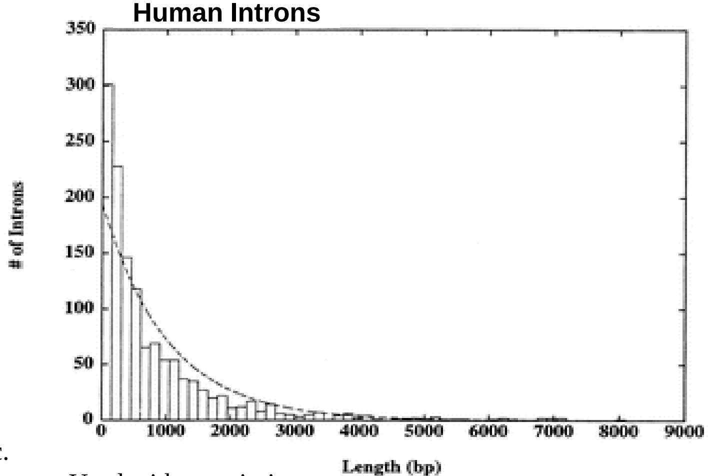
Generalized HMMs (GHMMs)

- GHMMs emit more than one symbol per state
- Emissions probabilities modeled by any arbitrary probabilistic model
- Feature lengths are explicitly modeled



W.H. Majaros (http://geneprediction.org/book/classroom.html)

Courtesy of William Majoros. Used with permission.



Burge, Karlin (1997)

Courtesy of Elsevier, Inc. http://www.sciencedirect.com. Used with permission.

GHMM Elements

- **States** Q
- **Observations** V
- **Initial state probabilities** $\pi_i = P(q_0=i)$
- **Transition Probabilities** $a_{jk}= P(q_i=k|q_{i-1}=j)$

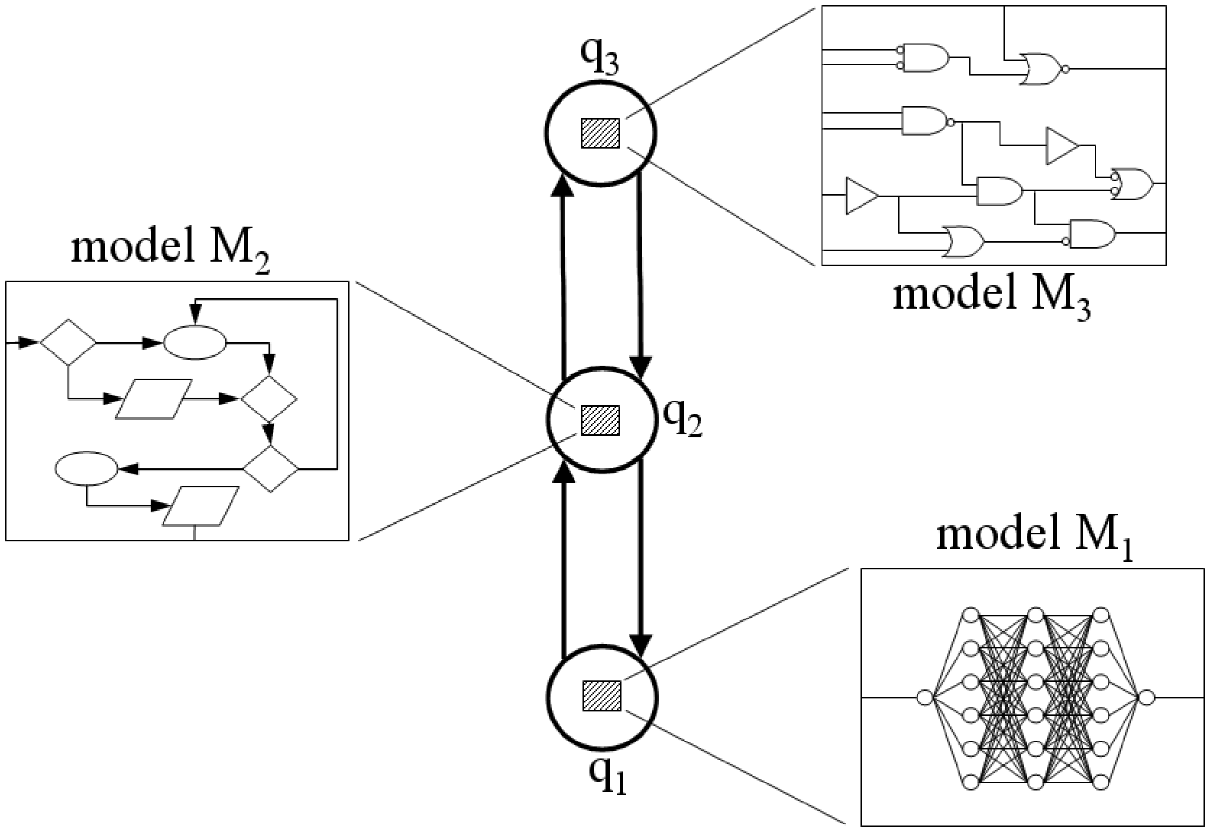
Like HMMs

- **Duration Probabilities** $f_k(d)=P(\text{state k of length d})$

- **Emission Probabilities** $e_k(X_{\alpha,\alpha+d})=P_k(X_\alpha \ldots X_{\alpha+d}| q_k,d)$

Now emit a subsequence

Model Abstraction in GHMMs



W.H. Majaros (http://geneprediction.org/book/classroom.html)

Models must return the probability of a subsequence given a state and duration

Courtesy of William Majoros. Used with permission.

GHMM Submodel Examples

1. WMM (Weight Matrix) $\prod_{i=0}^{L-1} P_i(x_i)$

2. Nth-order Markov Chain (MC) $\prod_{i=0}^{n-1} P(x_i \mid x_0 \ldots x_{i-1}) \prod_{i=n}^{L-1} P(x_i \mid x_{i-n} \ldots x_{i-1})$

3. Three-Periodic Markov Chain (3PMC) $\prod_{i=0}^{L-1} P_{(f+i)(\bmod 3)}(x_i)$

5. Codon Bias $\prod_{i=0}^{n-1} P(x_{\alpha+3i} x_{\alpha+3i+1} x_{\alpha+3i+2})$

6. MDD

Ref: Burge C (1997) Identification of complete gene structures in human genomic DNA. *PhD thesis*. Stanford University.

7. Interpolated Markov Model

$$P_e^{IMM}(s \mid g_0 \ldots g_{k-1}) = \begin{cases} \lambda_k^G P_e(s \mid g_0 \ldots g_{k-1}) + (1-\lambda_k^G) P_e^{IMM}(s \mid g_1 \ldots g_{k-1}) & \text{if } k > 0 \\ P_e(s) & \text{if } k = 0 \end{cases}$$

Ref: Salzberg SL, Delcher AL, Kasif S, White O (1998) Microbial gene identification using interpolated Markov models. *Nucleic Acids Research* 26:544-548.

http://geneprediction.org/book/classroom.html

Courtesy of William Majoros. Used with permission.

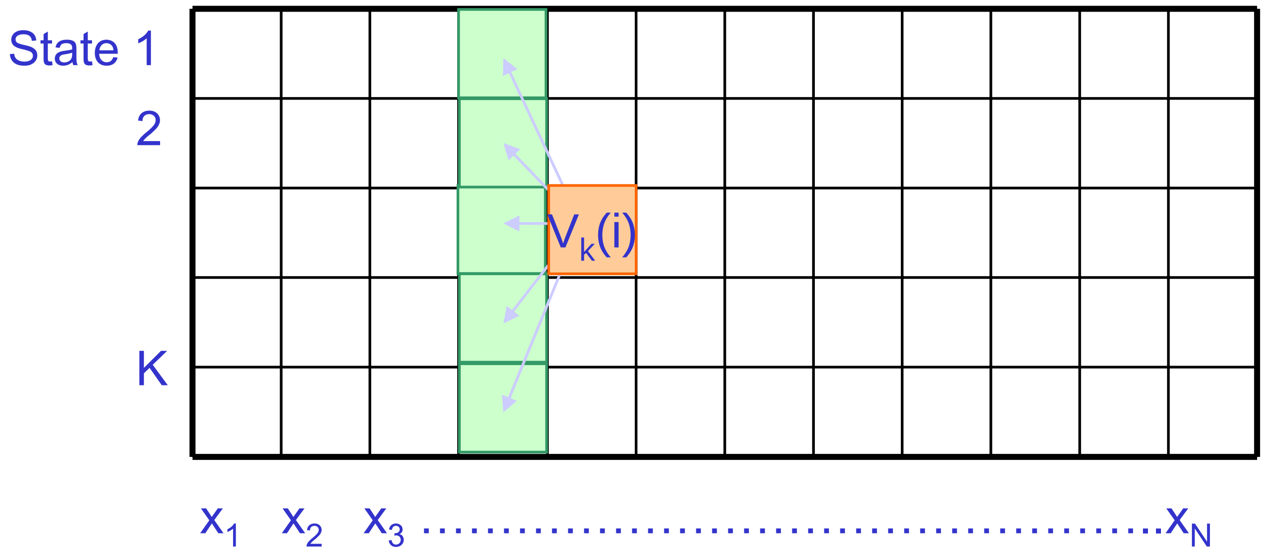
GHMMs as Generative Models

Like HMM, we can use a GHMM to generate a sequence <u>and</u> state labeling

- **Choose initial state, q_1, from π_i**
- **Choose a duration d_1, from length distribution $f_{q1}(d)$**
- **Generate a subsequence from state model $e_{q1}(X_{1,d})$**
- **Choose a subsequent state, q_2, conditioned on the current state, according to $a_{jk}=P(q_k|q_j)$**
- **Repeat until number of nucleotides equals desired length of sequence**

Given a sequence, how do we pick a state labeling (segmentation)?

The Viterbi Algorithm - HMMs



Input: x = x1……xN

Initialization:

$V_0(0)=1$, $V_k(0) = 0$, for all $k > 0$

Iteration:

$V_k(i) = e_K(x_i) \times \max_j a_{jk} V_j(i-1)$

Termination:

$P(x, \pi^*) = \max_k V_k(N)$

Traceback:

Follow max pointers back

In practice:

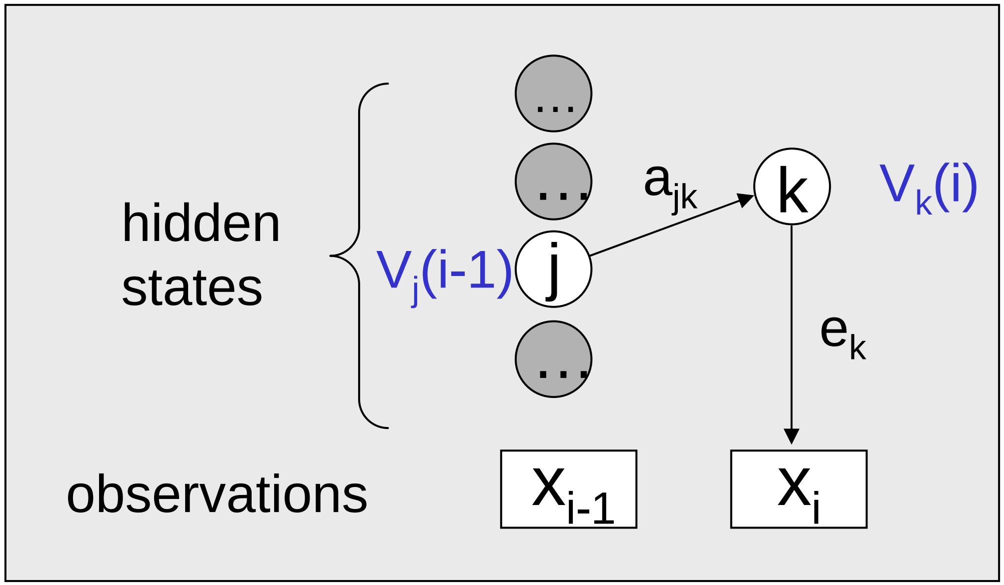
Use log scores for computation

Running time and space:

Time: $O(K^2N)$

Space: $O(KN)$

HMM Viterbi Recursion



- Assume we know V_j for the previous time step (i-1)
- Calculate $V_k(i) = e_k(x_i) \;*\; \max_j\left(V_j(i-1) \times a_{jk} \right)$

current max

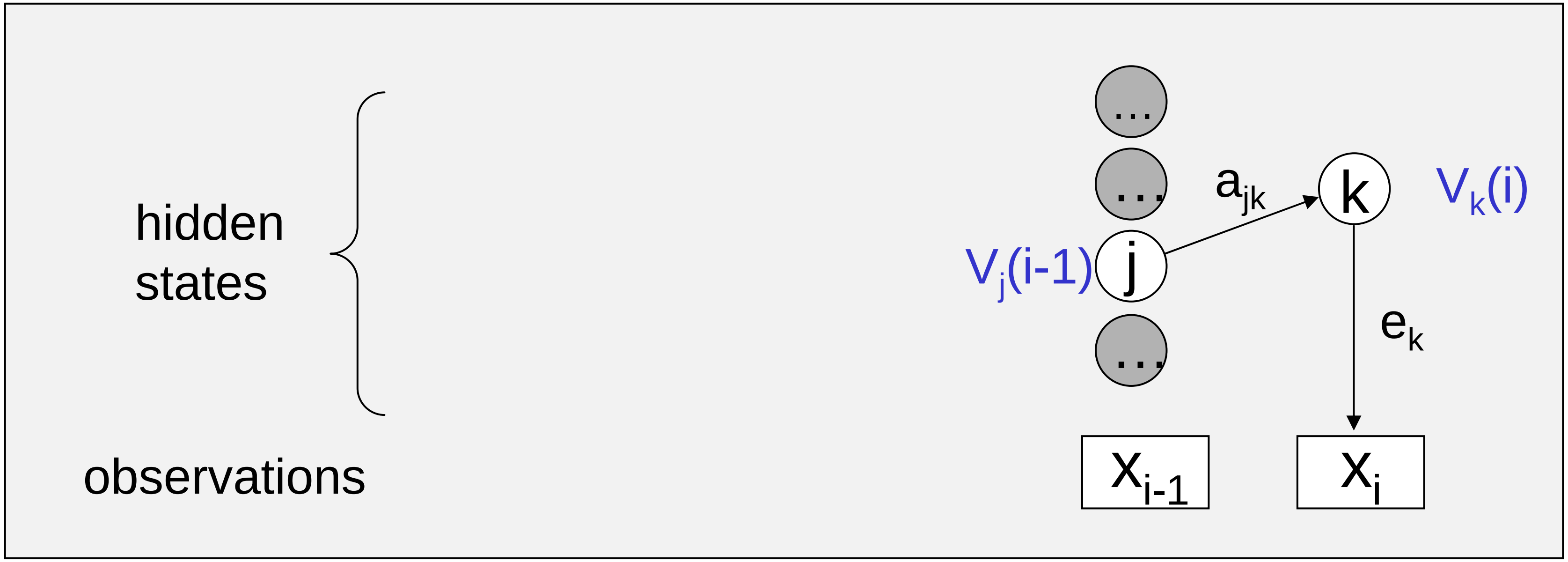
this emission

max ending in state j at step i-1

Transition from state j

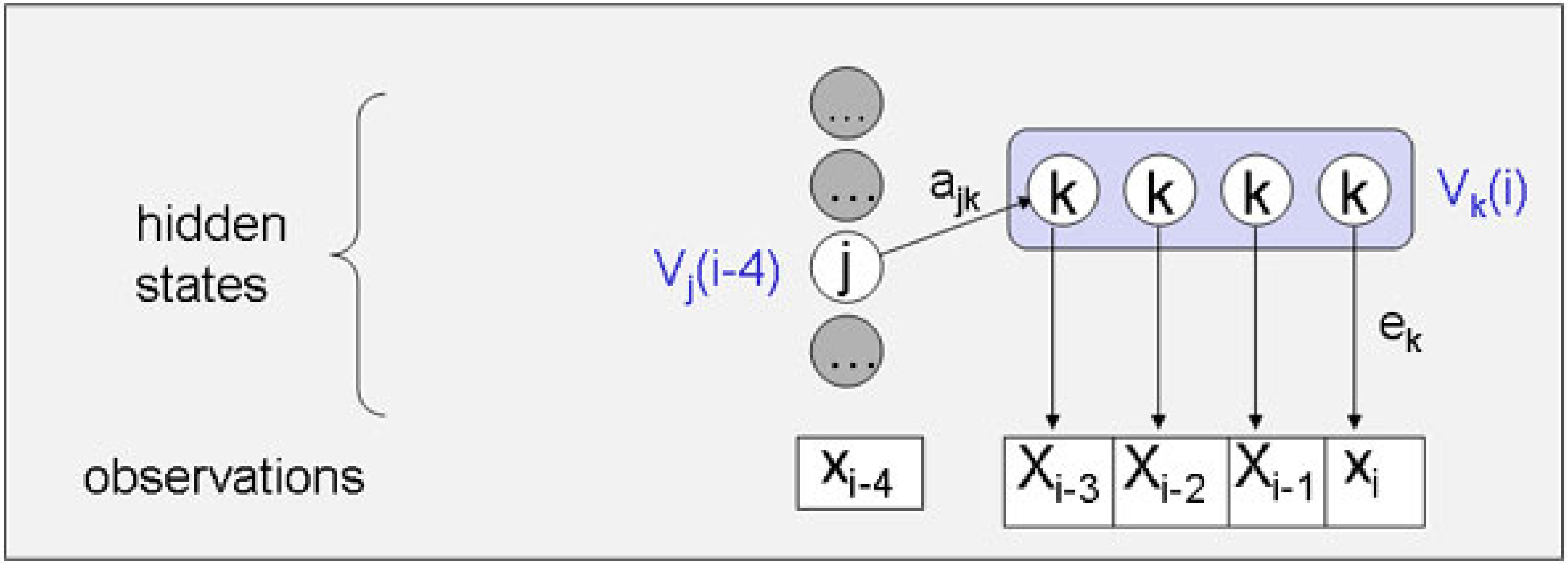
all possible previous states j

GHMM Recursion



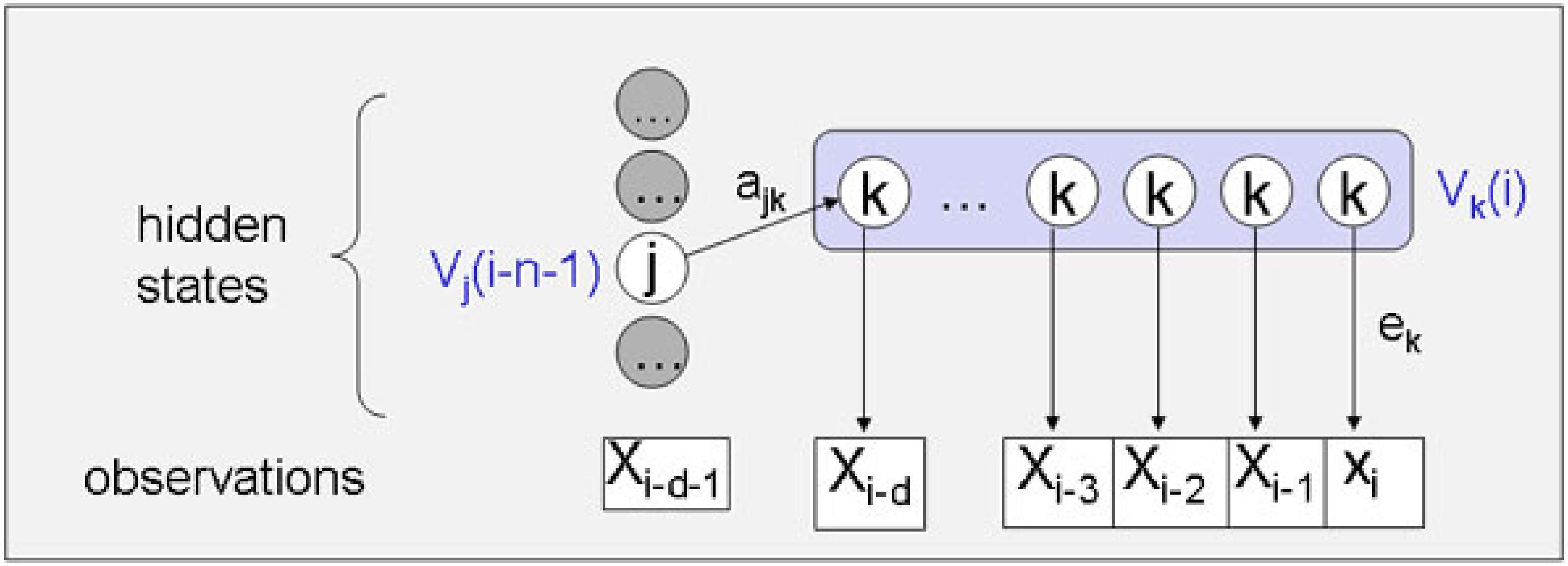
We could have come from state j at the last position…

GHMM Recursion



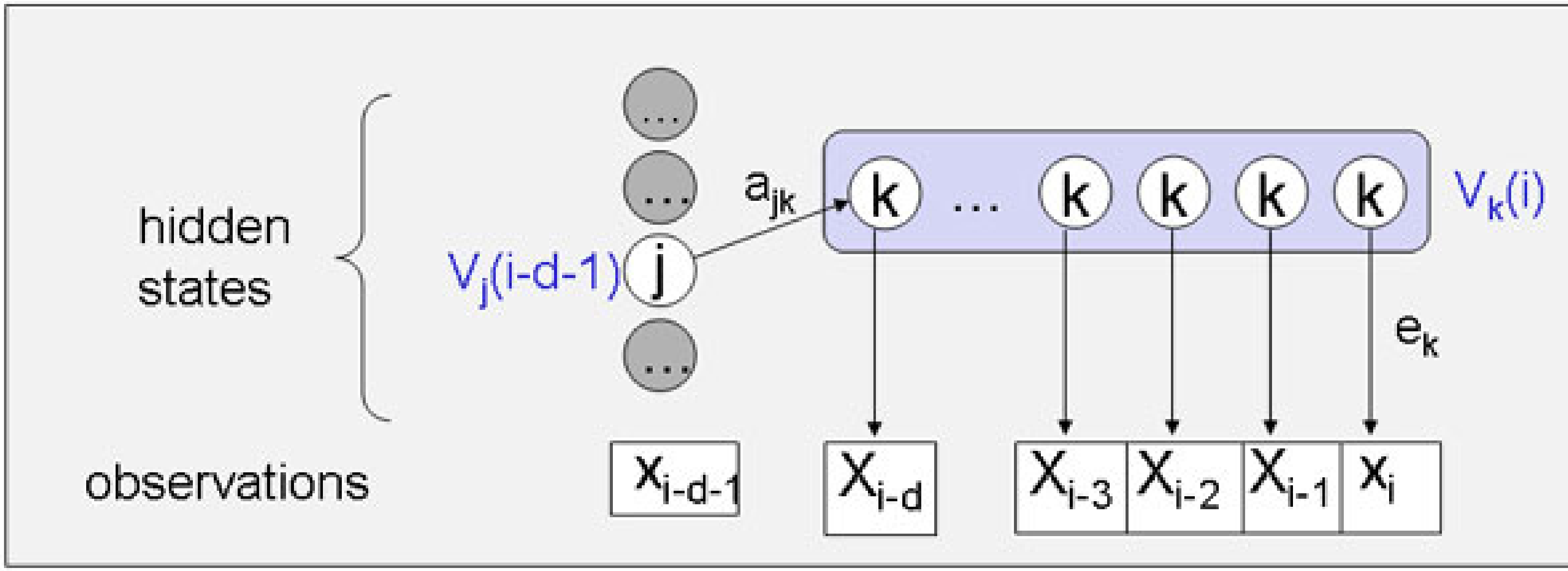
But we could also have come from state j 4 positions ago…
(State k could have duration 4 so far)

GHMM Recursion



In general, we could have come from state j d positions ago
(State k could have duration d so far)

GHMM Recursion



This leads to the following recursion equation:

$$V_k(i) = \underbrace{\max_j \max_d}_{\text{Max over all prev states } \textit{and} \text{ state durations}} \left[P\left(x_i \ldots x_{i-d} \mid k\right) \cdot V_j(i-d) \cdot P\left(d \middle| k\right) \cdot a_{jk} \right]$$

Max over all prev states *and* state durations

Prob of subsequence given state k

Max ending in state j at step i-d

Prob that state k has duration d

Transition from j->k

Comparing GHMMs and HMMs

HMM – $O(K^2L)$

$$V_k(i) = e_k(x_i) * \boxed{\max_j \left(V_j(i-1) \times a_{jk} \right)}$$

current max — this emission — max ending in state j at step i-1 — Transition from state j

all possible previous states j

GHMM – $O(K^2L^3)$

Similar modifications needed for forward and backward algorithms

$$V_k(i) = \underbrace{\max_j \max_d}_{} \left[P\left(x_i \ldots x_{i-d} \mid k\right) \cdot V_j(i-d) \cdot P\left(d \middle| k\right) \cdot a_{jk} \right]$$

Max over all prev states *and* state durations

Prob of subsequence given state k

Max ending in state j at step i-d

Prob that state k has duration d

Transition from j->k

J. Mol. Biol. (1997) 268, 78–94

JMB

Prediction of Complete Gene Structures in Human Genomic DNA

Chris Burge* and Samuel Karlin

Department of Mathematics
Stanford University, Stanford
CA, 94305, USA

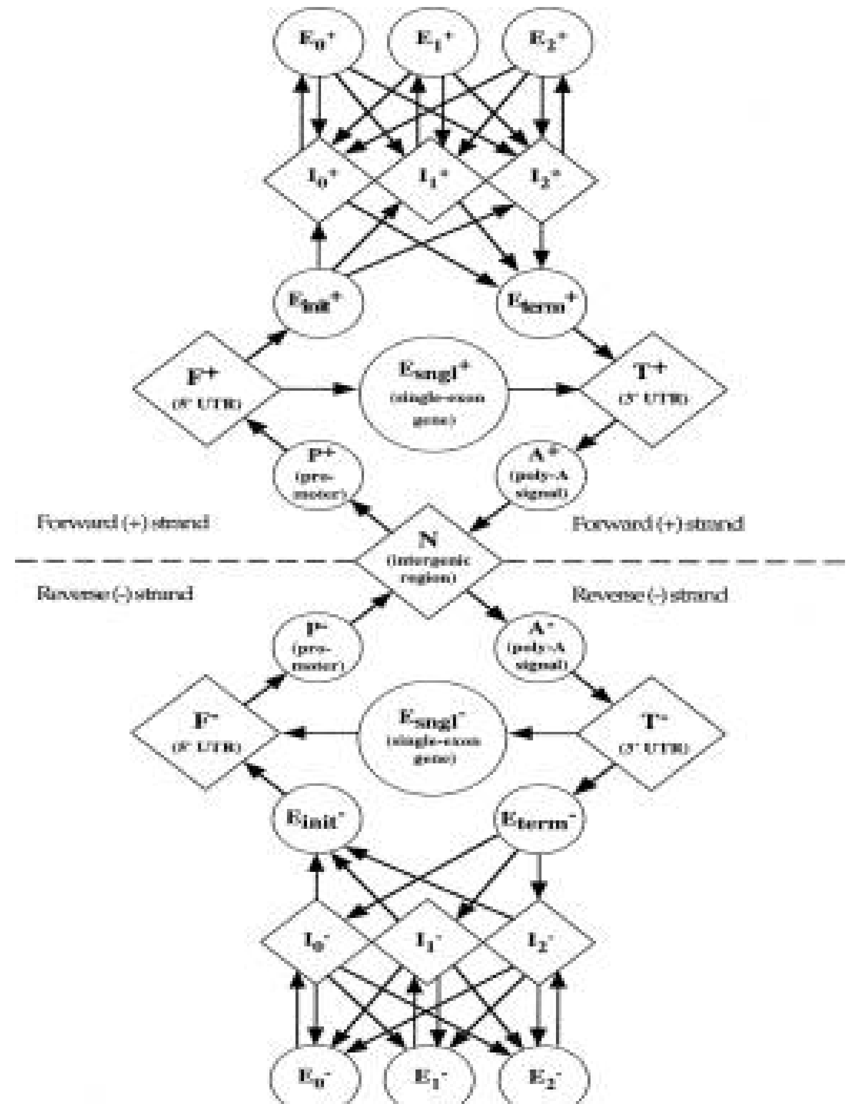
We introduce a general probabilistic model of the gene structure of human genomic sequences which incorporates descriptions of the basic transcriptional, translational and splicing signals, as well as length distributions and compositional features of exons, introns and intergenic regions. Distinct sets of model parameters are derived to account for the many substantial differences in gene density and structure observed in distinct C + G compositional regions of the human genome. In addition, new models of the donor and acceptor splice signals are described which capture potentially important dependencies between signal positions. The model is applied to the problem of gene identification in a computer program, GENSCAN, which identifies complete exon/intron structures of genes in genomic DNA. Novel features of the program include the capacity to predict multiple genes in a sequence, to deal with partial as well as complete genes, and to predict consistent sets of genes occurring on either or both DNA strands. GENSCAN is shown to have substantially higher accuracy than existing methods when tested on standardized sets of human and vertebrate genes, with 75 to 80% of exons identified exactly. The program is also capable of indicating fairly accurately the reliability of each predicted exon. Consistently high levels of accuracy are observed for sequences of differing C + G content and for distinct groups of vertebrates.

© 1997 Academic Press Limited

Courtesy of Elsevier, Inc. http://www.sciencedirect.com. Used with permission.

Genscan - Burge and Karlin, (1997)

- Explicit State Duration GHMM
- 5^{th} order markov models for coding and non-coding sequences
- Each CDS frame has own model
- WAM models for start/stop codons and acceptor sites
- MDD model for donor sites
- Separate parameters for regions of different GC content
- Model +/- strand concurrently

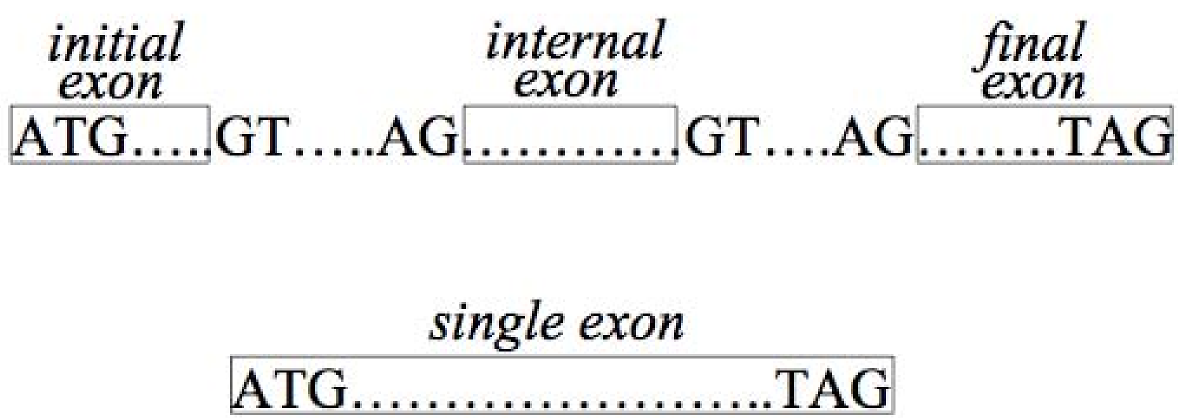


Courtesy of Elsevier, Inc. http://www.sciencedirect.com. Used with permission.

Types of Exons

Three types of exons are defined, for convenience:

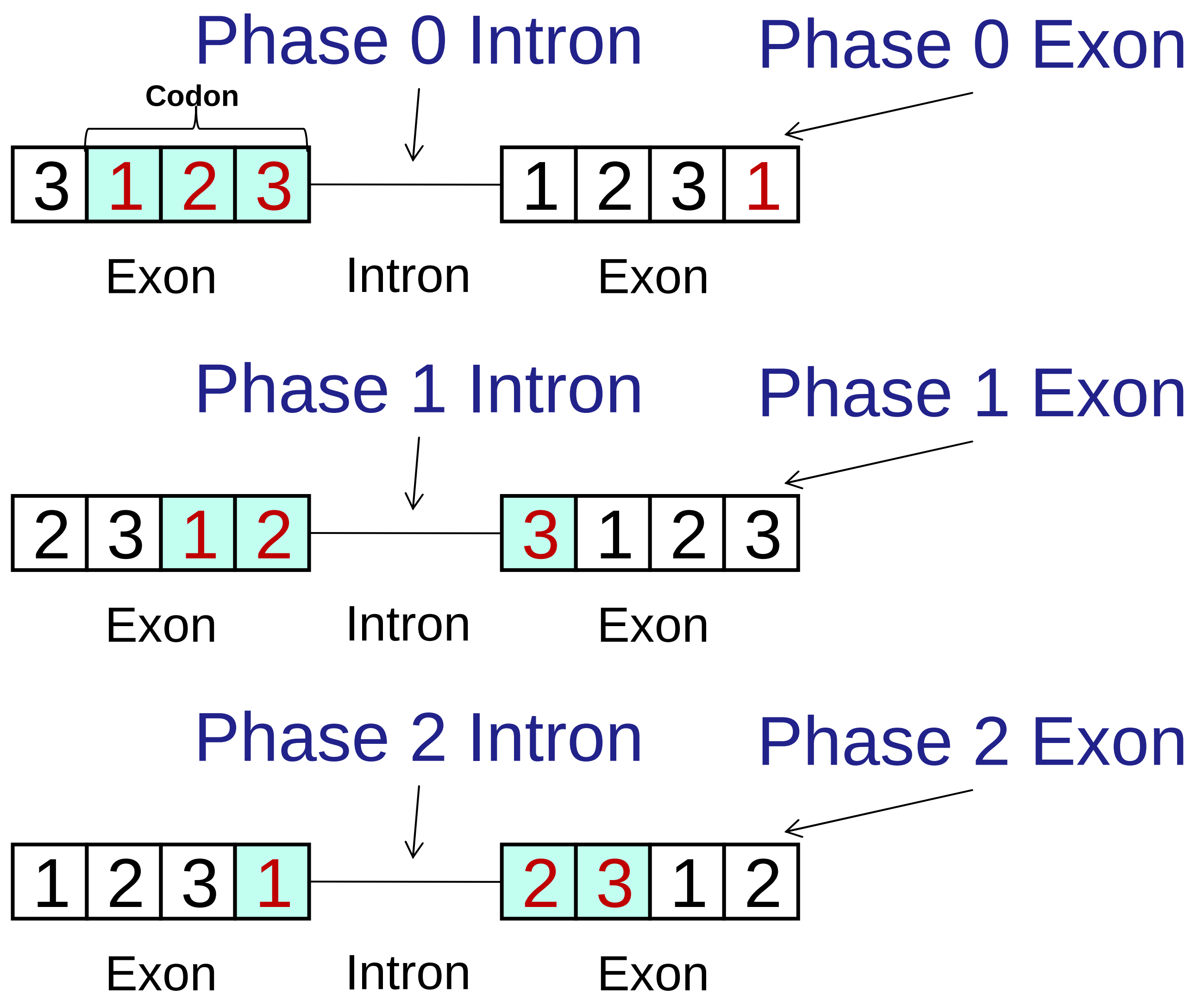
- *initial exons* extend from a start codon to the first donor site;
- *internal exons* extend from one acceptor site to the next donor site;
- *final exons* extend from the last acceptor site to the stop codon;
- *single exons* (which occur only in *intronless genes*) extend from the start codon to the stop codon:



Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

Intron and Exon Phase

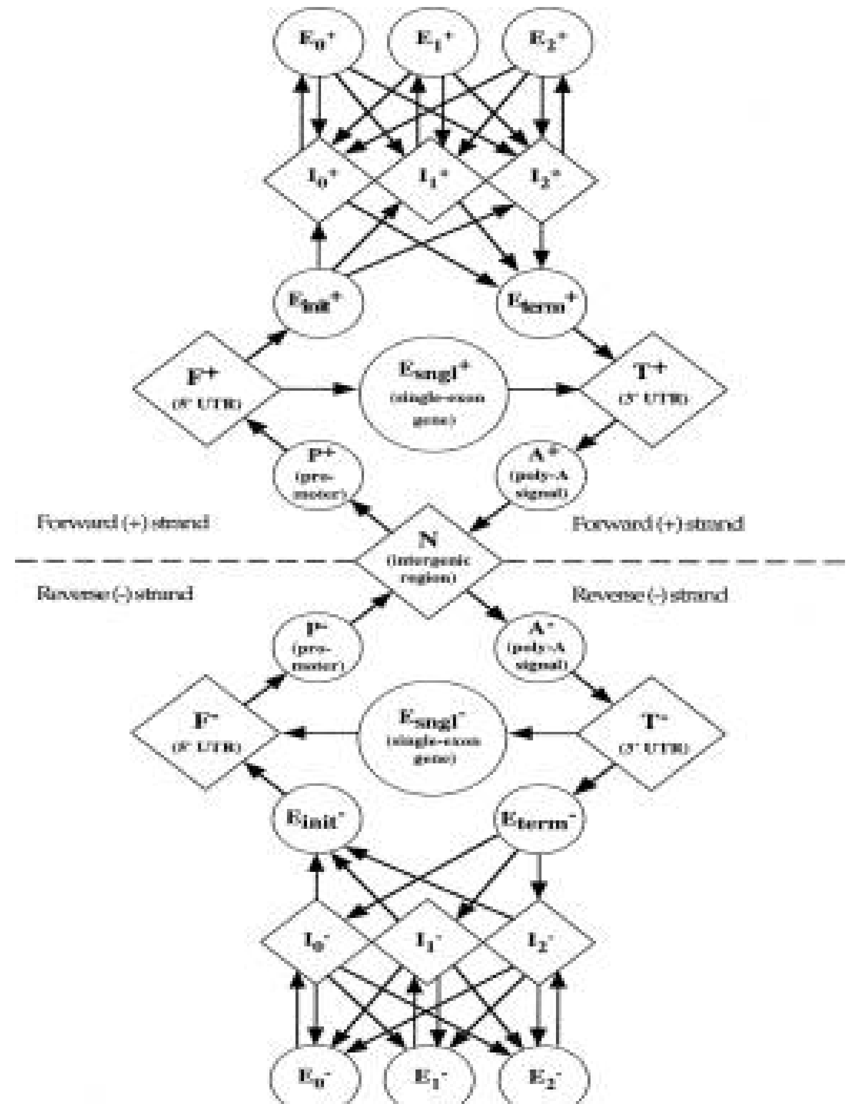


Two State Types in Genscan

- D-type represented by diamonds
- C-type represented by circles

D states are always followed by C and vice versa

C states always preceded by <u>same</u> D-state



Courtesy of Elsevier, Inc. http://www.sciencedirect.com. Used with permission.

Two State Types in Genscan

- D-Type – geometric length distribution
  - Intergenic regions
  - UTRs
  - Introns

Sequence models are "factorable":

$$P_k\left(Xa,c\right) = P_k(Xa,b)P_k(Xb+1,c)$$

$$f_k(d) = P(\text{state duration d}|\text{state k}) = \mathrm{p_k}\, f_k(d-1)$$

Increasing duration by one changes probability by constant factor

Two State Types in Genscan

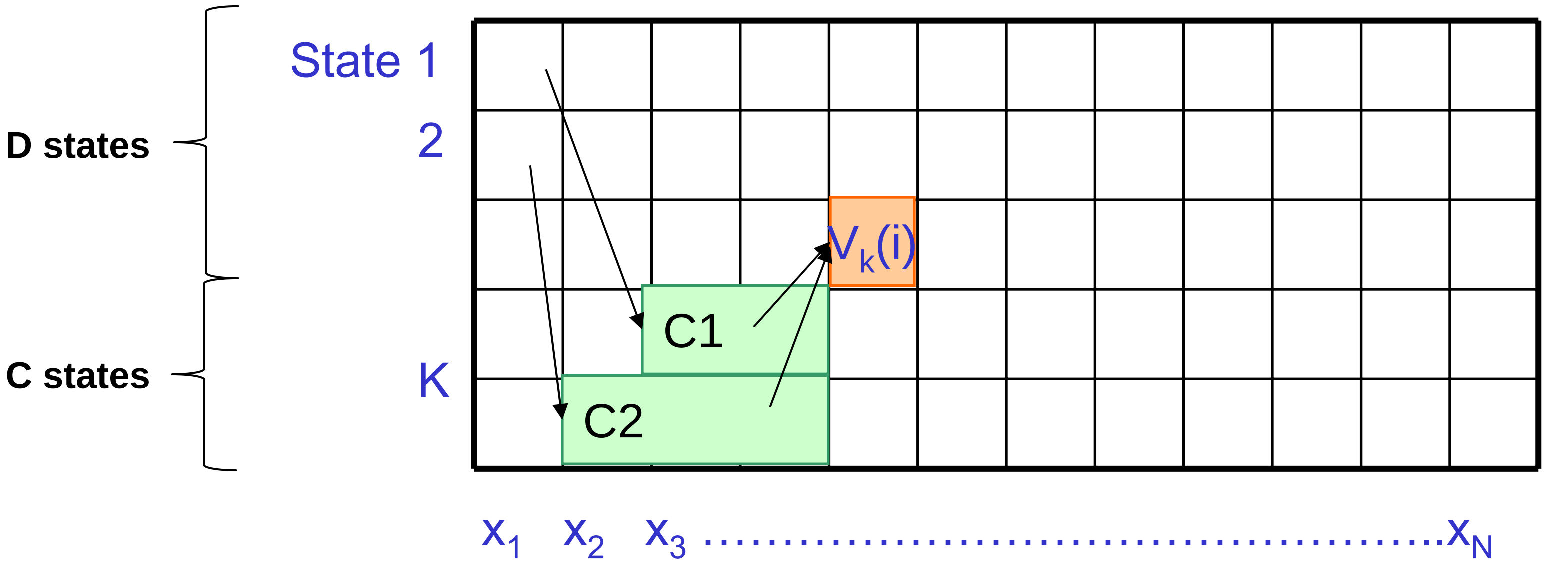
- C-Type – general length distributions and sequence generating modes
  - Exons (initial, internal, terminal)
  - Promotors
  - Poly-A Signals

Genscan Inference

- Genscan uses same basic forward, backward, and viterbi algorithms as generic GHMMs

- But assumptions about C, and D states reduce algorithmic complexity

Genscan Inference – C-state List



$L_k(i)$

C1, duration 2, previous D-state=1
C2, duration 3, previous D-state=2

Genscan Viterbi Induction

Max from previous step in this state

Extend D-type state by one step (factorable state)

Probability of emiting one more nucleotide from state k

$$V_k(i+1) = \max\ [\ V_k(i) \cdot p_k \cdot e_k(X_{i+1}),$$

Was in state K in previous step

$$\max_{c\ \in\ \text{c-type states}} \{V_{y_c}(i-d-1) \cdot \mathrm{P}(y_c|c)(1\text{-}p_{y_c}) \bullet \mathrm{P}(d|c)\mathrm{P}(X_{i\text{-}d}..X_i|c) \bullet \mathrm{P}(k|c) \cdot p_k \cdot e_k(X_{i+1})\}]$$

Just transitioned from c-type state c of duration d which previously transitioned from D-type state y_c

Max probability of ending D type state y_c at position i-d-1

Probability of transition from y_c to c

Terminate D-type state length

Probability that C state was duration d

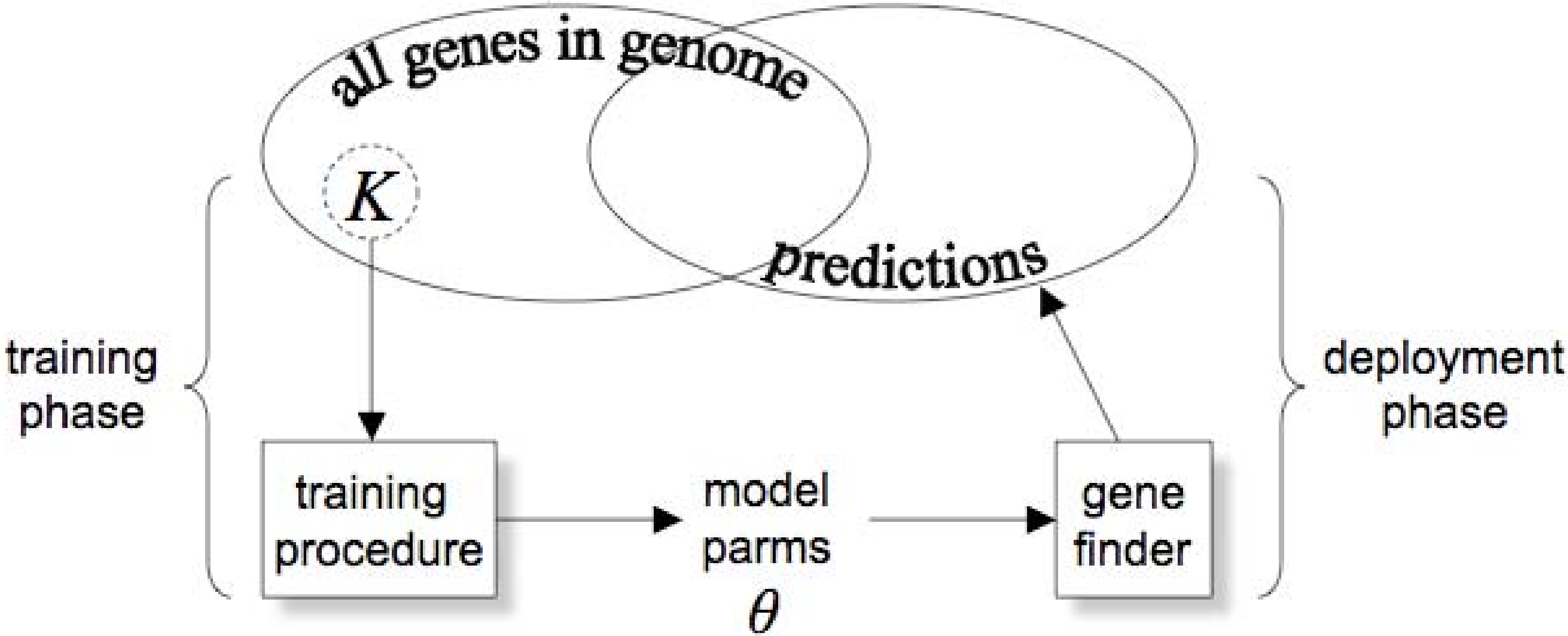
Probability of subsequence of length d from state c

Probability of transition from c to k

Probability of nucleotide from state k

Training A Gene Predictor

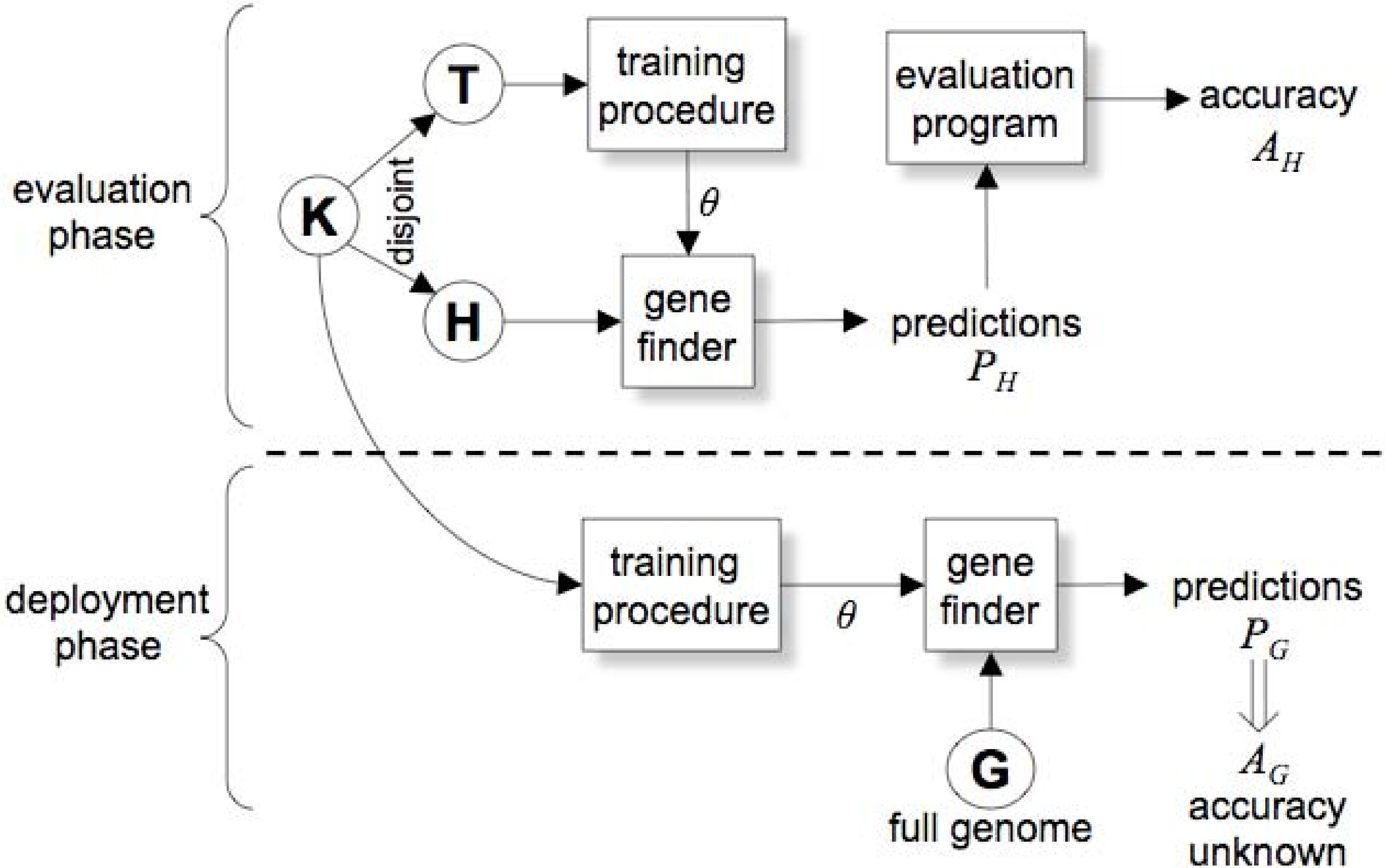
During *training* of a gene finder, only a subset *K* of an organism's gene set will be available for training:



The gene finder will later be *deployed* for use in predicting the rest of the organism's genes. The way in which the *model parameters* are inferred during training can significantly affect the accuracy of the deployed program.

Courtesy of William Majoros. Used with permission. http://geneprediction.org/book/classroom.html

Training A Gene Predictor

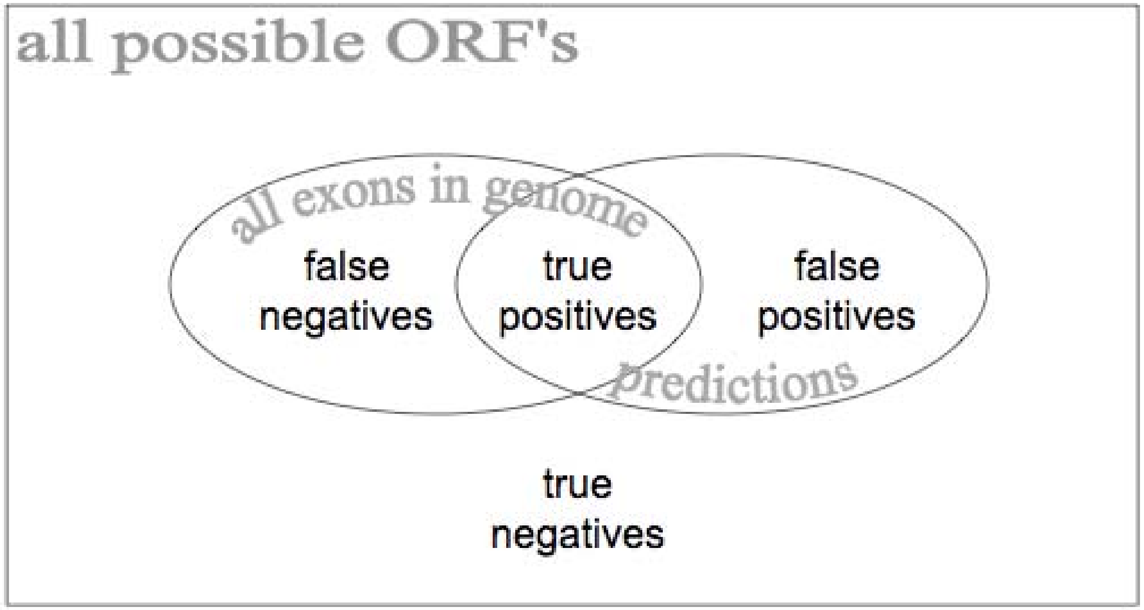


Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

Gene Prediction Accuracy

Gene predictions can be evaluated in terms of *true positives* (predicted features that are real), *true negatives* (non-predicted features that are not real), *false positives* (predicted features that are not real), and *false negatives* (real features that were not predicted:



These definitions can be applied at the *whole-gene*, *whole-exon*, or *individual nucleotide* level to arrive at three sets of statistics.

Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

Accuracy Metrics

$$Sn = \frac{TP}{TP+FN} \qquad Sp = \frac{TP}{TP+FP} \qquad F = \frac{2 \times Sn \times Sp}{Sn + Sp}$$

$$SMC = \frac{TP+TN}{TP+TN+FP+FN}$$

$$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP+FN) \times (TN+FP) \times (TP+FP) \times (TN+FN)}}.$$

$$ACP = \frac{1}{n}\left(\frac{TP}{TP+FN} + \frac{TP}{TP+FP} + \frac{TN}{TN+FP} + \frac{TN}{TN+FN}\right),$$

$$AC = 2(ACP - 0.5).$$

Courtesy of William Majoros. Used with permission.

http://geneprediction.org/book/classroom.html

More Information

- http://genes.mit.edu/burgelab/links.html
- http://www.geneprediction.org/book/classroom.html