

CSE 549: Models for ab initio Gene Finding



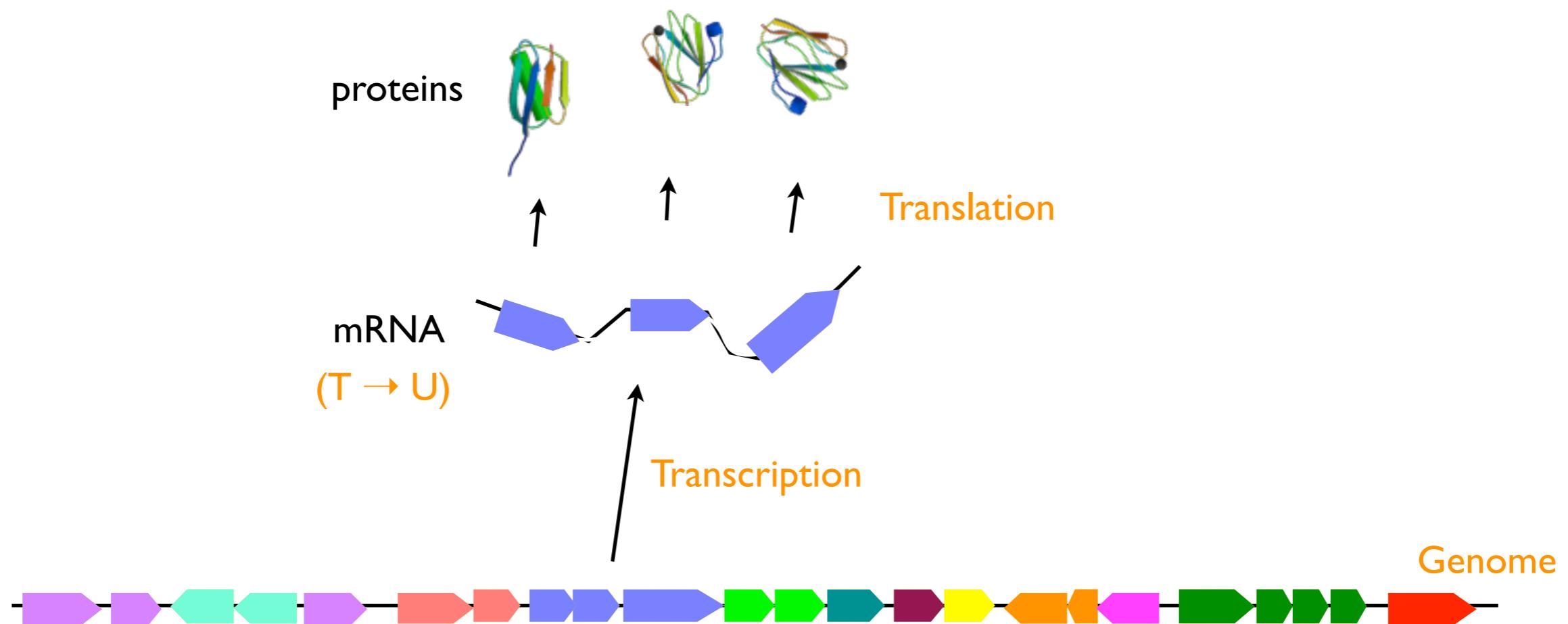
Genome of the Cow

a sequence of 2.86 billion letters

enough letters to fill a million pages of a typical book.

```
TATGGAGCCAGGTGCCTGGGGCAACAAGACTGTGGTCACTGAATTCATCCTTCTGGTCTAACAGAGAACATAG  
AACTGCAATCCATCCTTTGCCATCTCCTCTTGCTATGTGATCACAGTCGGGGCAACTGAGTATCCTG  
GCCGCCATCTTGTGGAGCCAAACTCCACACCCCCATGTACTACTCCTGGGGAACCTTCTGCTGGACAT  
TGGGTGCATCACTGTCACCATTCCCAGCTGGCCTGTCTGACCCACCAATGCCGGTCCCTATGCAG  
CCTGCATCTCACAGCTCTTCCACCTCCTGGCTGGAGTGGACTGTCACCTCCTGACAGCCATGGCCTAC  
GACCGCTACCTGGCCATTGCCAGCCCCACCTATAGCATCCGCATGAGCCGTGACGTCCAGGGAGCCCTGGT  
GGCCGTCTGCTGCTCCATCTCCTCATCAATGCTCTGACCCACACAGTGGCTGTCTGCTGGACTTCTGCG  
GCCCTAACGTGGTCAACCACTTCTACTGTGACCTCCGCCCTTCCAGCTCTGCTCCAGCATCCACCTC  
AACGGGCAGCTACTTCTGGGGGCCACCTCATGGGGTGGTCCCCATGGTCTTCATCTCGGTATCCTATGC  
CCACGTGGCAGCCGCAGTCCTGCGGATCCGCTGGCAGAGGGCAGGAAGAAAGCCTCTCCACGTGTGGCTCCC  
ACCTCACCGTGGTCTGCATCTTATGGAACCGGCTTCTCAGCTACATGCGCCTGGCTCCGTCCGCCTCA  
GACAAGGACAAGGGCATTGGCATCCTAACACTGTCATCAGCCCCATGCTGAACCCACTCATACAGCCTCCG  
GAACCCTGATGTGCAGGGGCCCTGAAGAGGTTGCTGACAGGGAAGCAGGGGGGGAGTG ...
```

“Central Dogma” of Biology



DNA =

- double-stranded, linear molecule
- each strand is string over {A,C,G,T}
- strands are complements of each other ($A \leftrightarrow T$; $C \leftrightarrow G$)
- substrings encode for genes (blue arrow) most of which encode for proteins



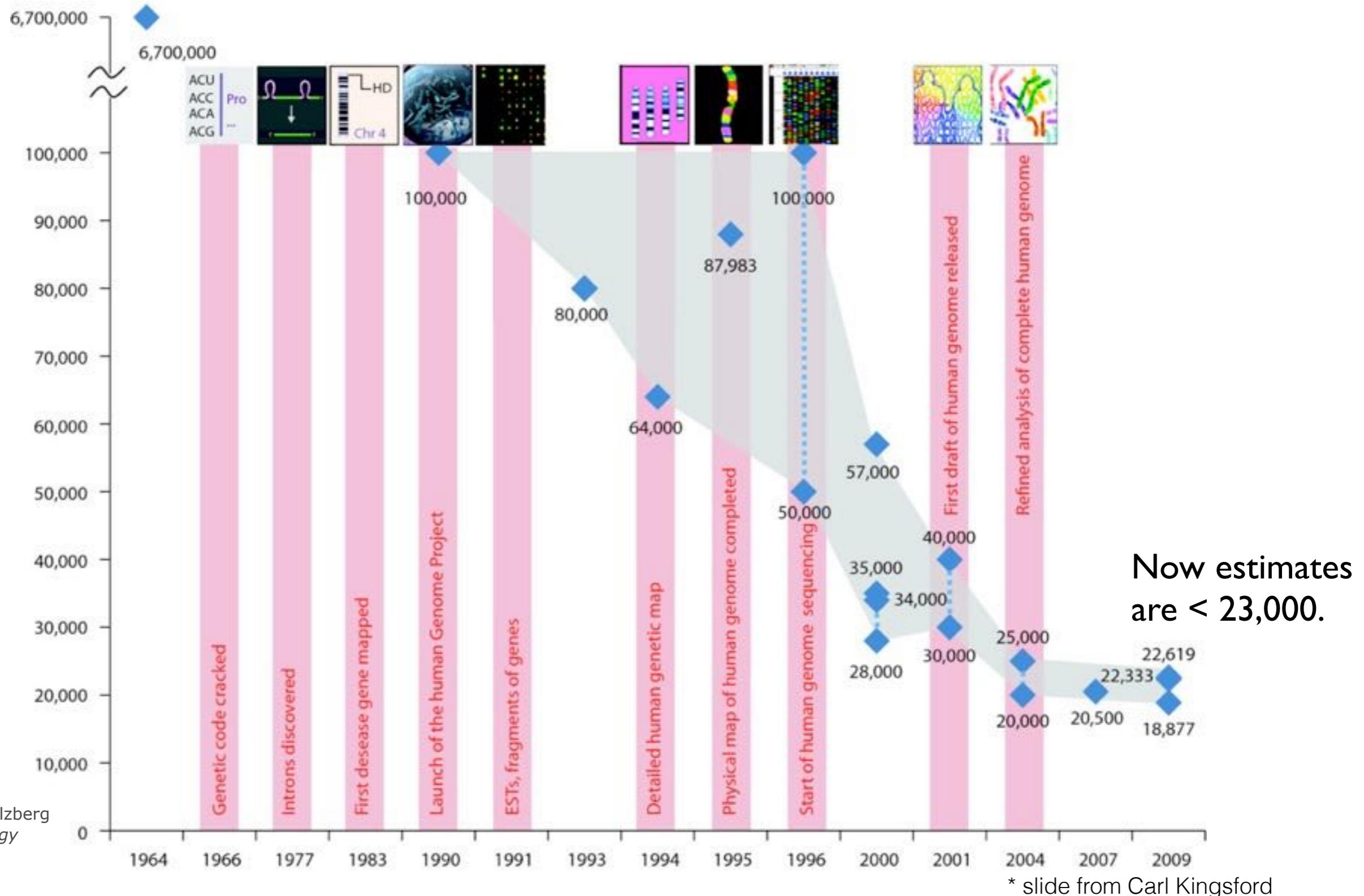
	2nd base								
	U		C		A		G		
1st base	U	UUU	(Phe/F) Phenylalanine	UCU	(Ser/S) Serine	UAU	(Tyr/Y) Tyrosine	UGU	(Cys/C) Cysteine
		UUC	(Phe/F) Phenylalanine	UCC	(Ser/S) Serine	UAC	(Tyr/Y) Tyrosine	UGC	(Cys/C) Cysteine
		UUA	(Leu/L) Leucine	UCA	(Ser/S) Serine	UAA	Ochre Stop	UGA	Opal Stop
		UUG	(Leu/L) Leucine	UCG	(Ser/S) Serine	UAG	Amber Stop	UGG	(Trp/W) Tryptophan
	C	CUU	(Leu/L) Leucine	CCU	(Pro/P) Proline	CAU	(His/H) Histidine	CGU	(Arg/R) Arginine
		CUC	(Leu/L) Leucine	CCC	(Pro/P) Proline	CAC	(His/H) Histidine	CGC	(Arg/R) Arginine
		CUA	(Leu/L) Leucine	CCA	(Pro/P) Proline	CAA	(Gln/Q) Glutamine	CGA	(Arg/R) Arginine
		CUG	(Leu/L) Leucine	CCG	(Pro/P) Proline	CAG	(Gln/Q) Glutamine	CGG	(Arg/R) Arginine
	A	AUU	(Ile/I) Isoleucine	ACU	(Thr/T) Threonine	AAU	(Asn/N) Asparagine	AGU	(Ser/S) Serine
		AUC	(Ile/I) Isoleucine	ACC	(Thr/T) Threonine	AAC	(Asn/N) Asparagine	AGC	(Ser/S) Serine
		AUA	(Ile/I) Isoleucine	ACA	(Thr/T) Threonine	AAA	(Lys/K) Lysine	AGA	(Arg/R) Arginine
		AUG [A]	(Met/M) Methionine	ACG	(Thr/T) Threonine	AAG	(Lys/K) Lysine	AGG	(Arg/R) Arginine
	G	GUU	(Val/V) Valine	GCU	(Ala/A) Alanine	GAU	(Asp/D) Aspartic acid	GGU	(Gly/G) Glycine
		GUC	(Val/V) Valine	GCC	(Ala/A) Alanine	GAC	(Asp/D) Aspartic acid	GGC	(Gly/G) Glycine
		GUU	(Val/V) Valine	GCA	(Ala/A) Alanine	GAA	(Glu/E) Glutamic acid	GGA	(Gly/G) Glycine
		GUG	(Val/V) Valine	GCG	(Ala/A) Alanine	GAG	(Glu/E) Glutamic acid	GGG	(Gly/G) Glycine

The Genetic Code

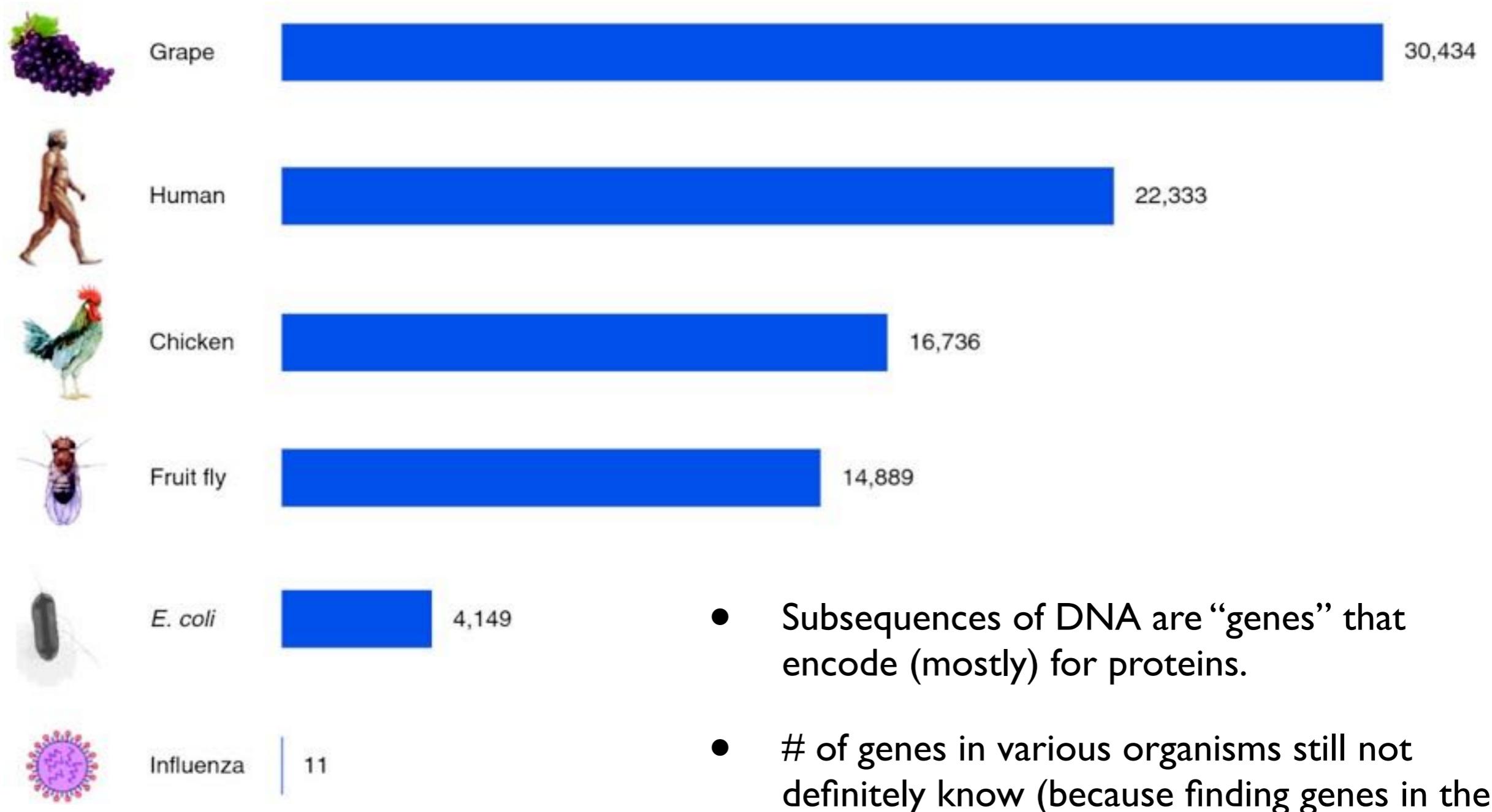
- There are 20 different amino acids & 64 different codons.
- Lots of different ways to encode for each amino acid.
- The 3rd base is typically less important for determining the amino acid
- Three different “stop” codons that signal the end of the gene
- Start codons differ depending on the organisms, but AUG is often used.

Estimates for the # of Human Genes

Before human genome sequence was available, many (but not all) estimates for # of genes were high (> 80,000).



of Genes in Various Organisms



Pertea and Salzberg
Genome Biology 2010
11:206

- Subsequences of DNA are “genes” that encode (mostly) for proteins.
- # of genes in various organisms still not definitely known (because finding genes in the sequence is a hard problem that we will talk about).
- But there are reasonably good estimates.

Finding Genes

We'll break gene finding methods into 3 main categories.

ab initio

latin — “from the beginning”
w/o experimental evidence

based on predictive modeling

how well do genomic
sequences score under
our “gene model”?

comparative

make use of knowledge
across species

a known human gene is
strong evidence for a
chimp gene

many “housekeeping” genes
are incredibly similar across
highly divergent species

combined / extrinsic

Make use of experimental
evidence (e.g. RNA-seq)

Evidence highlights
transcribed regions

Gene structure extracted
from evidence (potentially
combined with model
predictions)

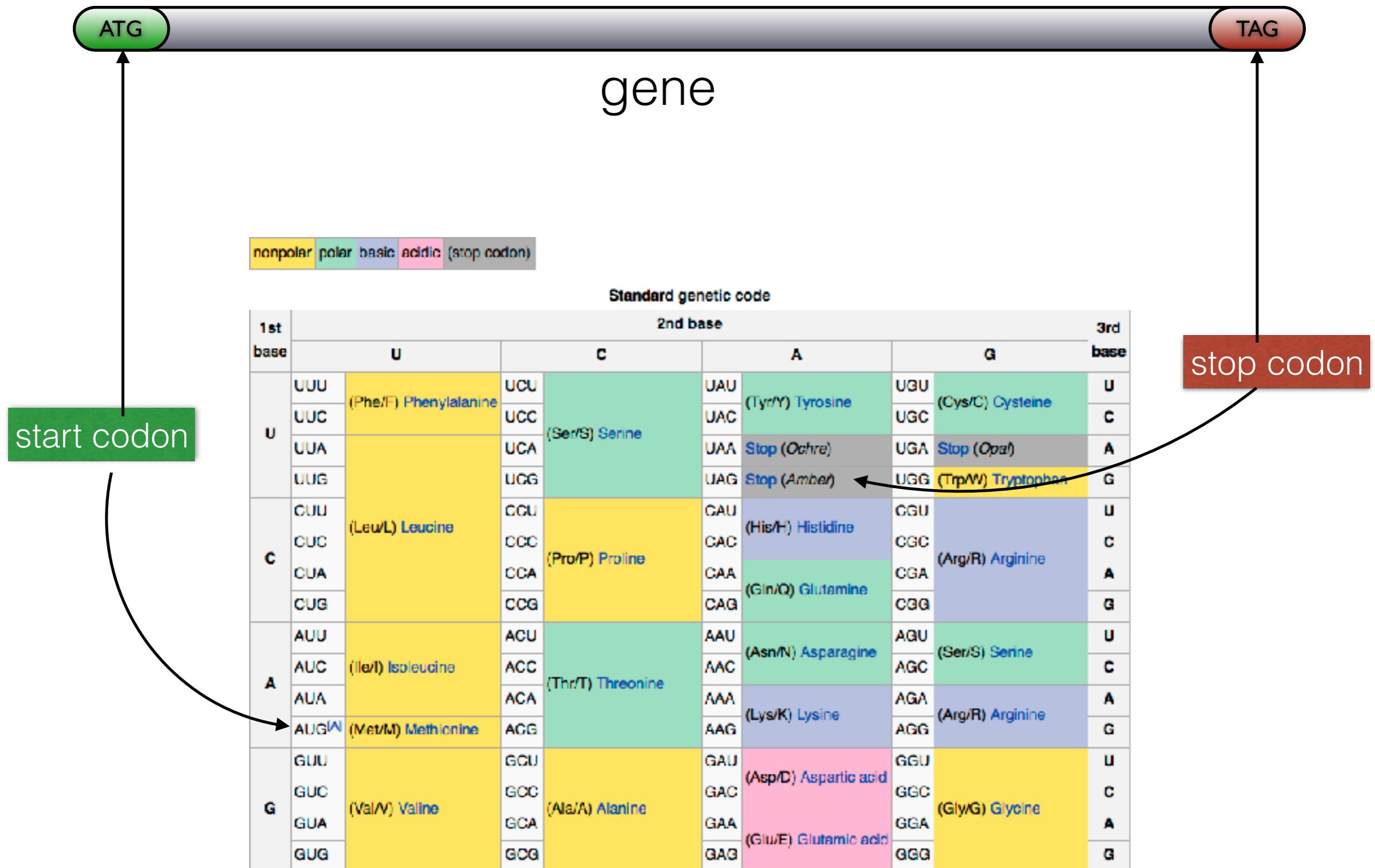
Ab Initio Prediction

Today, we'll focus mainly on *ab initio* prediction, but will touch on comparative prediction.

- How do we build models of genes?
- How do we use these models to predict genes?
- What features must the models capture?
- Where are the models good, where do they fail?

Prokaryotic Gene Prediction

Genes in prokaryotes *generally* have a simple structure

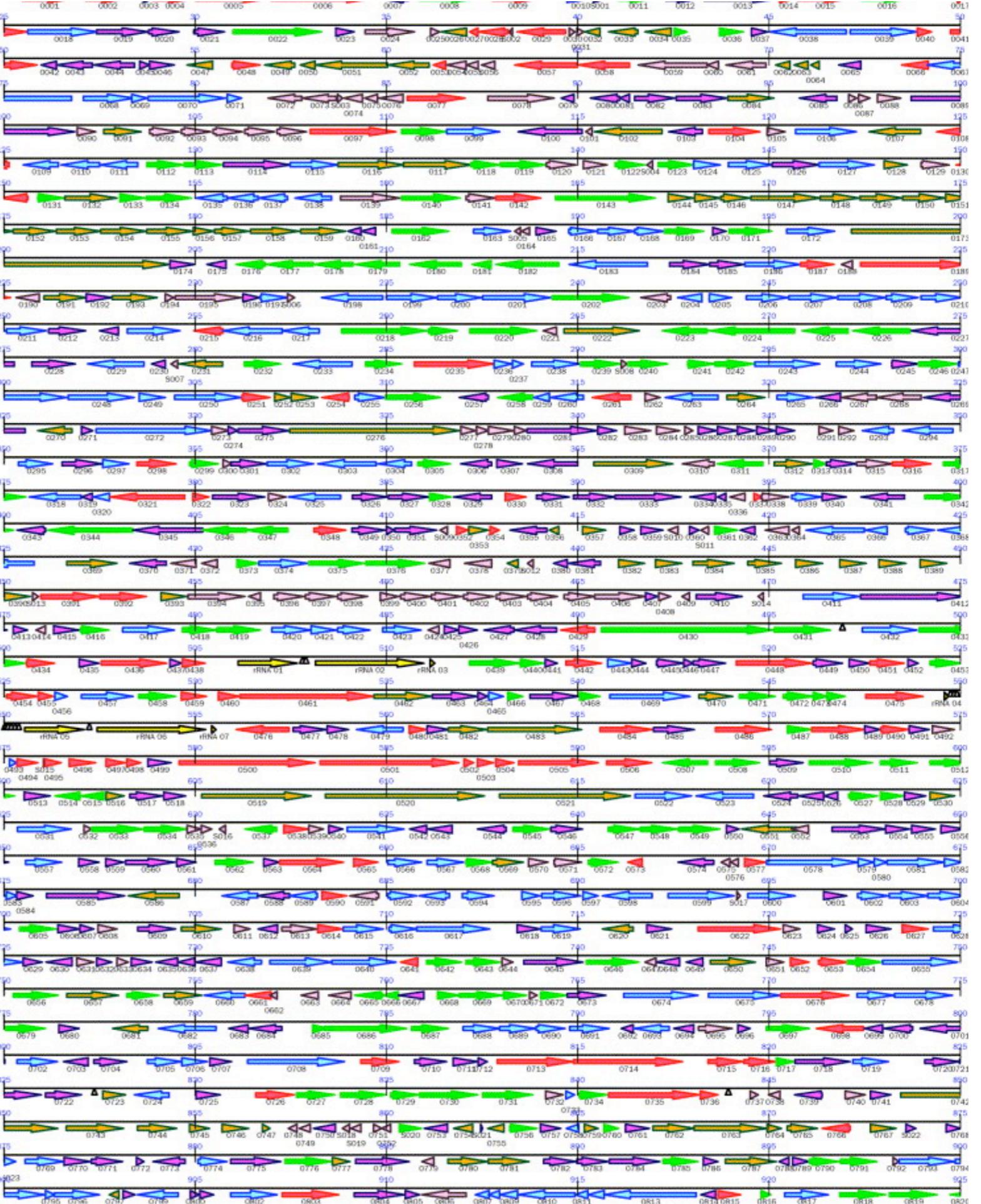


Bacterial genes

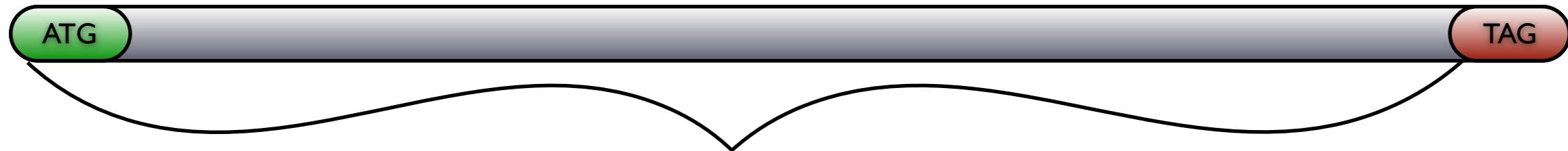
Genes (colored arrows)
packed tightly into the
Staphylococcus aureus
genome

In bacteria, one gene
corresponds to one
continuous interval on
the genome

We have good methods
for predicting where
these genes are



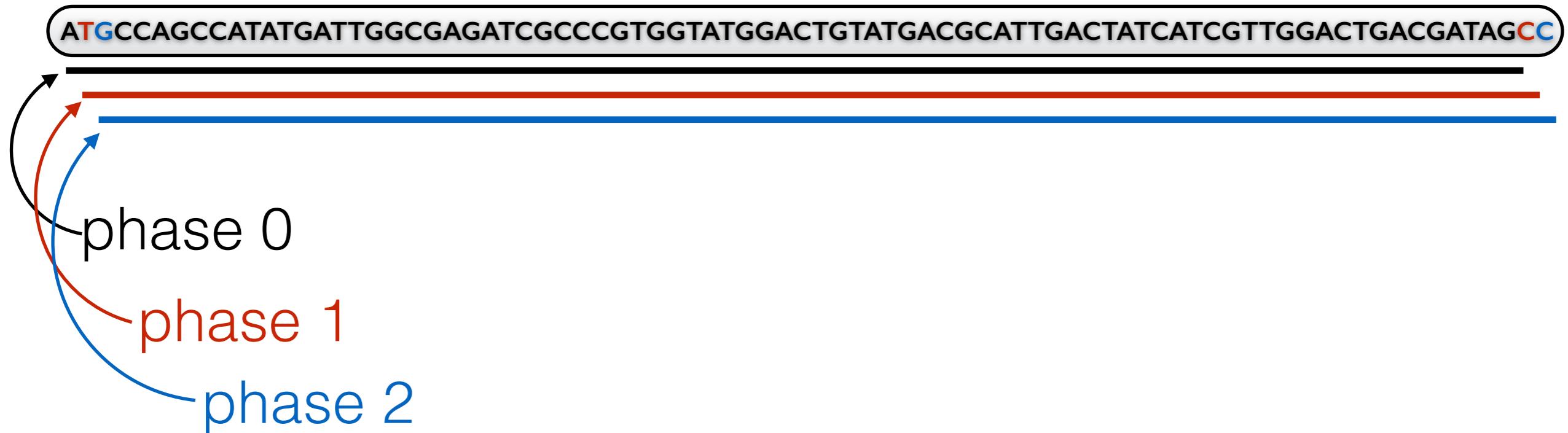
Prokaryotic Gene Prediction



We call the region from a start codon until a stop codon in the same “phase” an **open reading frame** (ORF).

ORFs are possible genes. All genes are ORFs, but all ORFs are **not genes**.

Prokaryotic Gene Prediction



The codons are read off as consecutive, disjoint sequences of 3 bases each. But we don't know where the first codon of a gene starts.

Each strand of DNA contains 3 “reading frames”. Consecutive, disjoint, codons in a gene are in the same “phase” (same index mod 3) as their predecessors.

Prokaryotic Gene Prediction

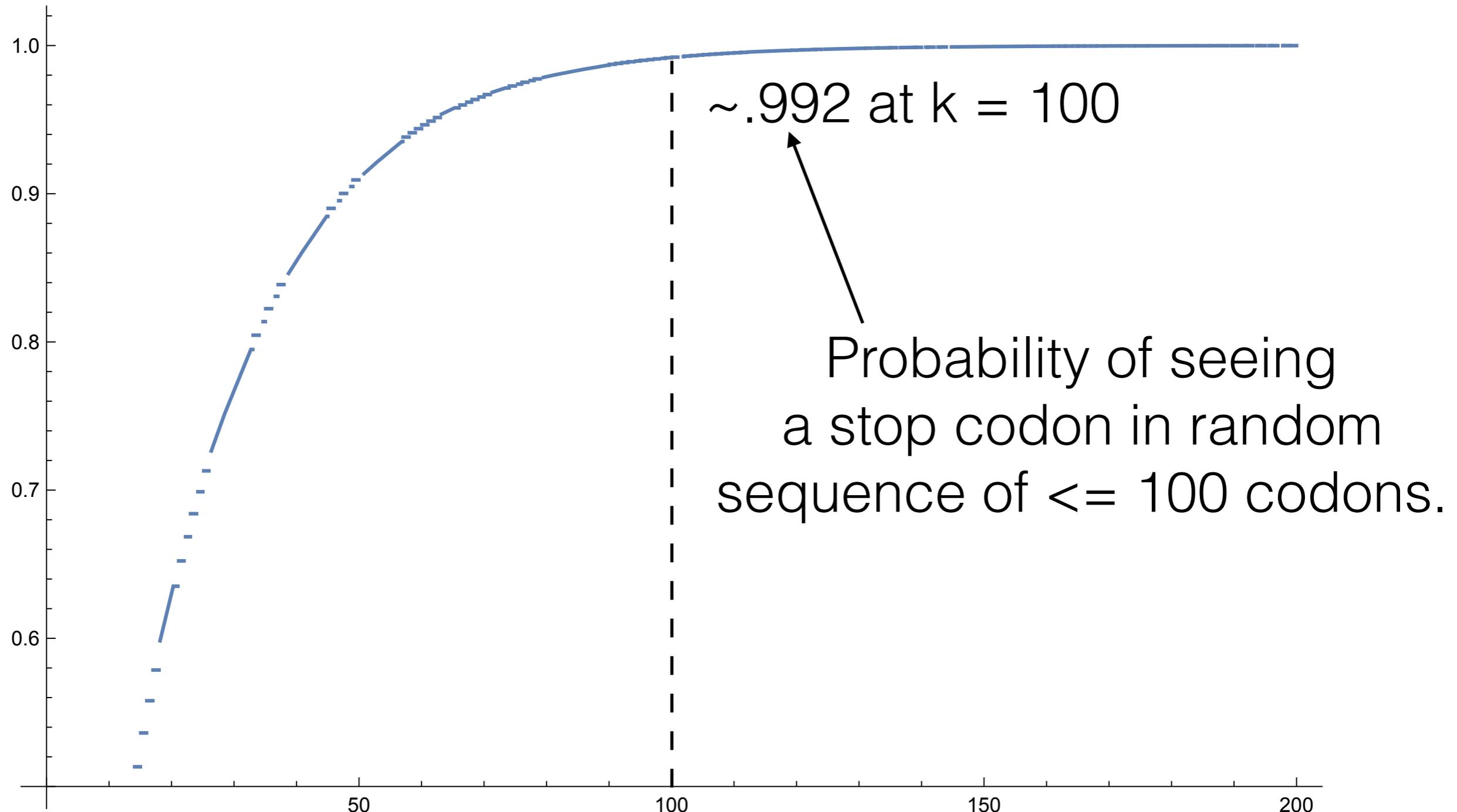


There's 1 start codon and 3 stop codons. In a random sequence, the probability of seeing 3 bases that represent a start codon is $1/64$. The prob. of seeing 3 bases that represent a stop codon is $3/64$.

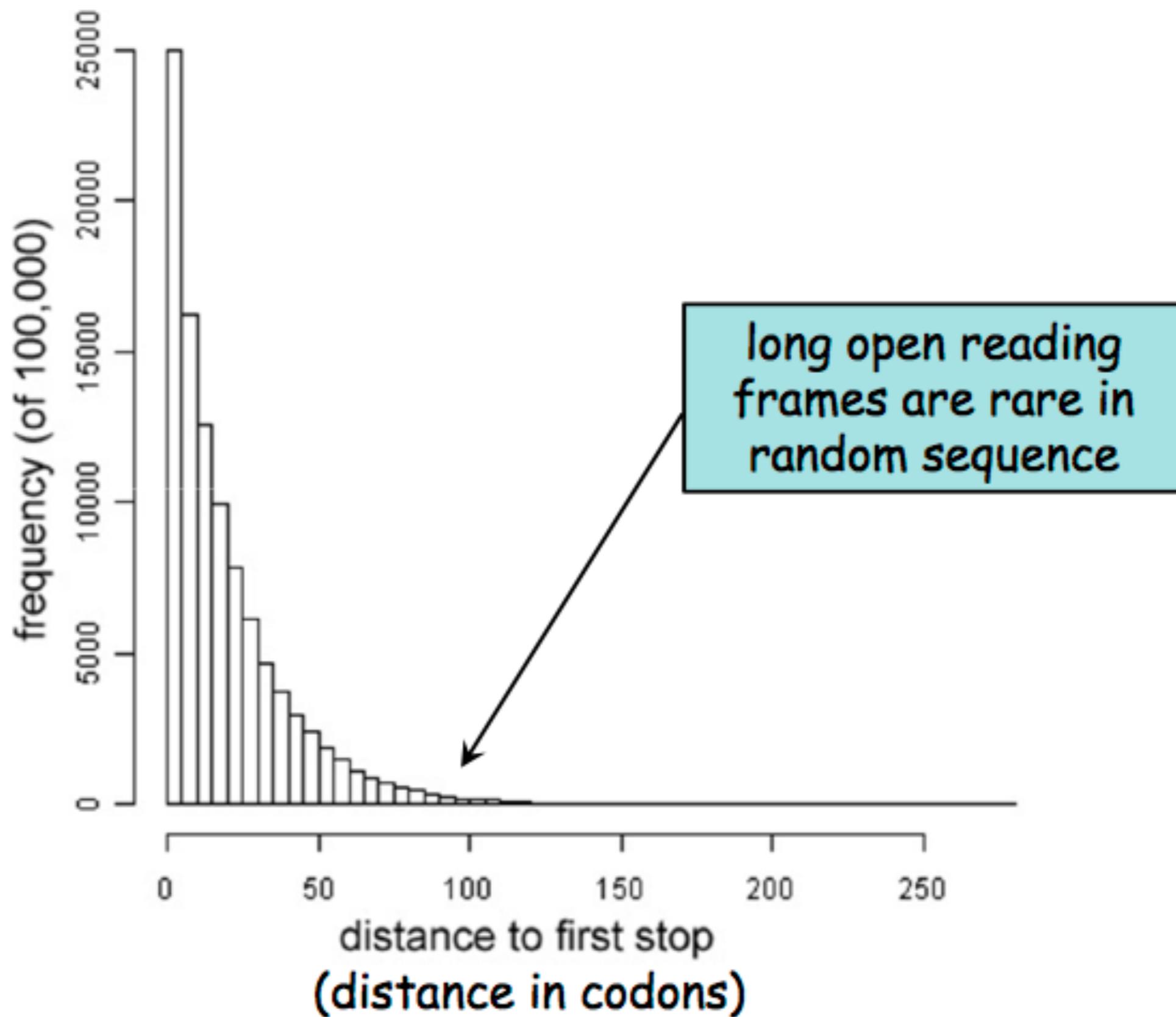
Probability of *not observing* a stop codon in a random sequence of length L decays geometrically in L .

Prokaryotic Gene Prediction

$$\text{CDF}(\text{Geom}(3/64)) = 1 - (1 - (3/64))^k$$



Prokaryotic Gene Prediction



Gene Prediction

So, even finding a long, non-interrupted ORF can be strong evidence that a stretch of sequence is a gene.

This leads to a **very simple** idea for a prokaryotic gene finding algorithm.

A Simple Gene Finder

1. Find all stop codons in genome
2. For each stop codon, find the in-frame start codon farthest upstream of the stop codon, without crossing another in-frame stop codon.

GGC **TAG** **ATG** AGG GCT CTA ACT **ATG** GGC GCG **TAA**

Each substring between the start and stop codons is called an **ORF**
“open reading frame”

3. Return the “long” ORF as predicted genes.

3 out of the 64 possible codons are stop codons \Rightarrow in random DNA,
every 22nd codon is expected to be a stop.

Gene Prediction

Another type of evidence is codon usage bias — a difference in the frequency of occurrence of synonymous codons in genic DNA.

Gene Prediction

Codon usage bias in anthrax

Bacillus anthracis str. A2012 [gbgbct]: 308 CDS's (64833 codons)

fields: [triplet] [frequency: per thousand] ([number])

UUU	34.7(2249)	UCU	15.7(1015)	UAU	34.5(2238)	UGU	6.8(441)
UUC	11.7(761)	UCC	4.5(293)	UAC	10.4(672)	UGC	2.6(167)
UUA	41.8(2709)	UCA	14.2(921)	UAA	2.8(184)	UGA	1.0(65)
UUG	12.9(835)	UCG	3.9(256)	UAG	0.9(59)	UGG	10.2(661)
CUU	14.5(937)	CCU	10.2(662)	CAU	16.7(1084)	CGU	11.0(715)
CUC	3.6(233)	CCC	2.5(165)	CAC	4.7(305)	CGC	2.9(188)
CUA	13.7(888)	CCA	12.6(817)	CAA	30.7(1993)	CGA	6.6(425)
CUG	5.0(326)	CCG	4.5(292)	CAG	10.3(671)	CGG	1.8(116)
AUU	43.5(2817)	ACU	13.7(886)	AAU	44.1(2856)	AGU	16.7(1082)
AUC	11.7(758)	ACC	5.1(328)	AAC	14.4(932)	AGC	6.2(403)
AUA	24.9(1616)	ACA	25.9(1678)	AAA	66.8(4332)	AGA	14.6(948)
AUG	25.3(1642)	ACG	8.6(555)	AAG	25.8(1671)	AGG	4.5(294)
GUU	19.5(1263)	GCU	17.0(1105)	GAU	38.9(2522)	GGU	17.6(1142)
GUC	5.4(347)	GCC	4.4(285)	GAC	9.1(589)	GGC	5.8(379)
GUA	26.1(1694)	GCA	22.0(1429)	GAA	55.1(3570)	GGA	19.7(1276)
GUG	11.0(711)	GCG	7.6(490)	GAG	19.7(1276)	GGG	9.5(614)

Coding GC 33.87% 1st letter GC 43.97% 2nd letter GC 30.99% 3rd letter GC 26.64%

Gene Finding as a Machine Learning Problem

- Given training examples of some known genes, can we distinguish ORFs that are genes from those that are not?
- **Idea:** can use distribution of codons to find genes.
 - every codon should be about equally likely in non-gene DNA.
 - every organism has a slightly different bias about how often certain codons are preferred.
 - could also use frequencies of longer strings (k-mers).

An Improved Simple Gene Finder

- Score each ORF using the product of the probability of each codon:

$$\text{GFScore}(g) = \Pr(\text{codon}_1) \times \Pr(\text{codon}_2) \times \Pr(\text{codon}_3) \times \dots \times \Pr(\text{codon}_n)$$

$$\log[\text{GFScore}(g)] = \log[\Pr(\text{codon}_1)] + \log[\Pr(\text{codon}_2)] + \dots + \log[\Pr(\text{codon}_n)]$$

But: as genes get longer, GFScore(g) will decrease.

One option: also score the sequence under a null / non-gene model NScore(g).

We can use the log-odds ratio of GFScore(g) to Score (G) to predict whether or not this ORF is a gene.

Glimmer

Salzberg et al., NAR, 1998

- Score ORFs using 6 Markov Models (not hidden):
 - 1 model for each reading frame (3 forward, 3 reverse)
- ORFs for which the highest scoring reading frame exceed some threshold are output as “putative” genes.
- Use “Interpolated Markov models” to adapt to data availability
- Handle overlapping ORFs

Interpolated HMMs

$$P(S|M) = \sum_{x=1}^n \text{IMM}_8(S_x)$$

Sequence → P(S|M) → Model

Length of the sequence → n

String ending at position x → S_x

$$P_i(S_x) = P(s_x|S_{x,i}) = \frac{f(S_{x,i})}{\sum_{b \in \{acgt\}} f(S_{x,i}, b)}$$

Sequence ending at pos x of length i — length i “context”

IMM score is a linear combination of 8th, 7th, ..., 0th order models:

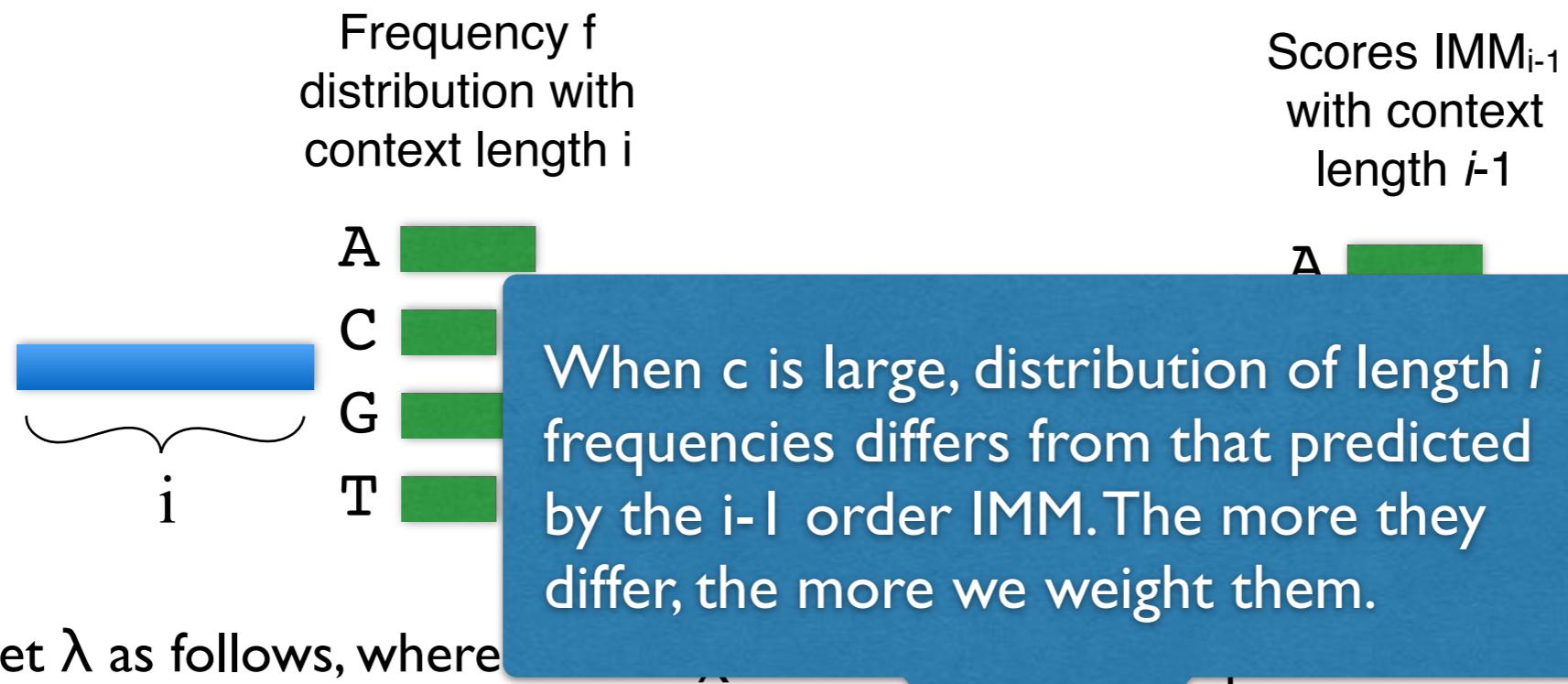
$$\text{IMM}_k(S_x) = \boxed{\lambda_k(S_{x-1})} \bullet P_k(S_x) + [1 - \boxed{\lambda_k(S_{x-1})}] \bullet \text{IMM}_{k-1}(S_x)$$

Weight of the kth-order context ending at position x-1

Probability of letter at position x from a kth-order model

Setting Parameters

- If # of occurrences of context k-mer ≥ 400 , $\lambda = 1$
- Otherwise compare the following with a χ^2 statistic:



- Set λ as follows, where $c = \sum_{b \in \{acgt\}} f(s_1 s_2 \dots s_i b)$ comes from the IMM distribution:

$$\lambda_i(S_{x-1}) = \begin{cases} 0.0 & \text{if } c < 0.50 \\ \frac{c}{400} \sum_{b \in \{acgt\}} f(s_1 s_2 \dots s_i b) & \text{if } c \geq 0.50 \end{cases}$$

IMM vs. 5th Order Markov Model

Model	Genes found	Genes missed	Additional genes
GLIMMER IMM	1680 (97.8%)	37	209
5 th -Order Markov	1574 (91.7%)	143	104

The first column indicates how many of the 1717 annotated genes in *H.influenzae* were found by each algorithm. The ‘additional genes’ column shows how many extra genes, not included in the 1717 annotated entries, were called genes by each method.

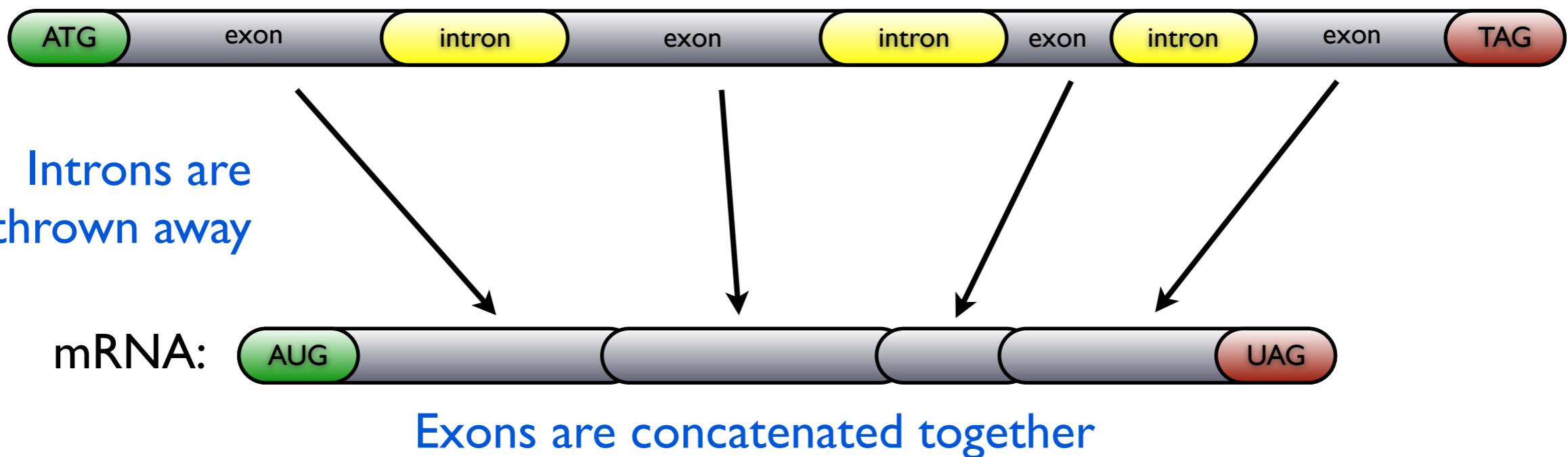
Salzberg et al., NAR, 1998

Eukaryotic Genes & Exon Splicing

Prokaryotic (bacterial) genes look like this:

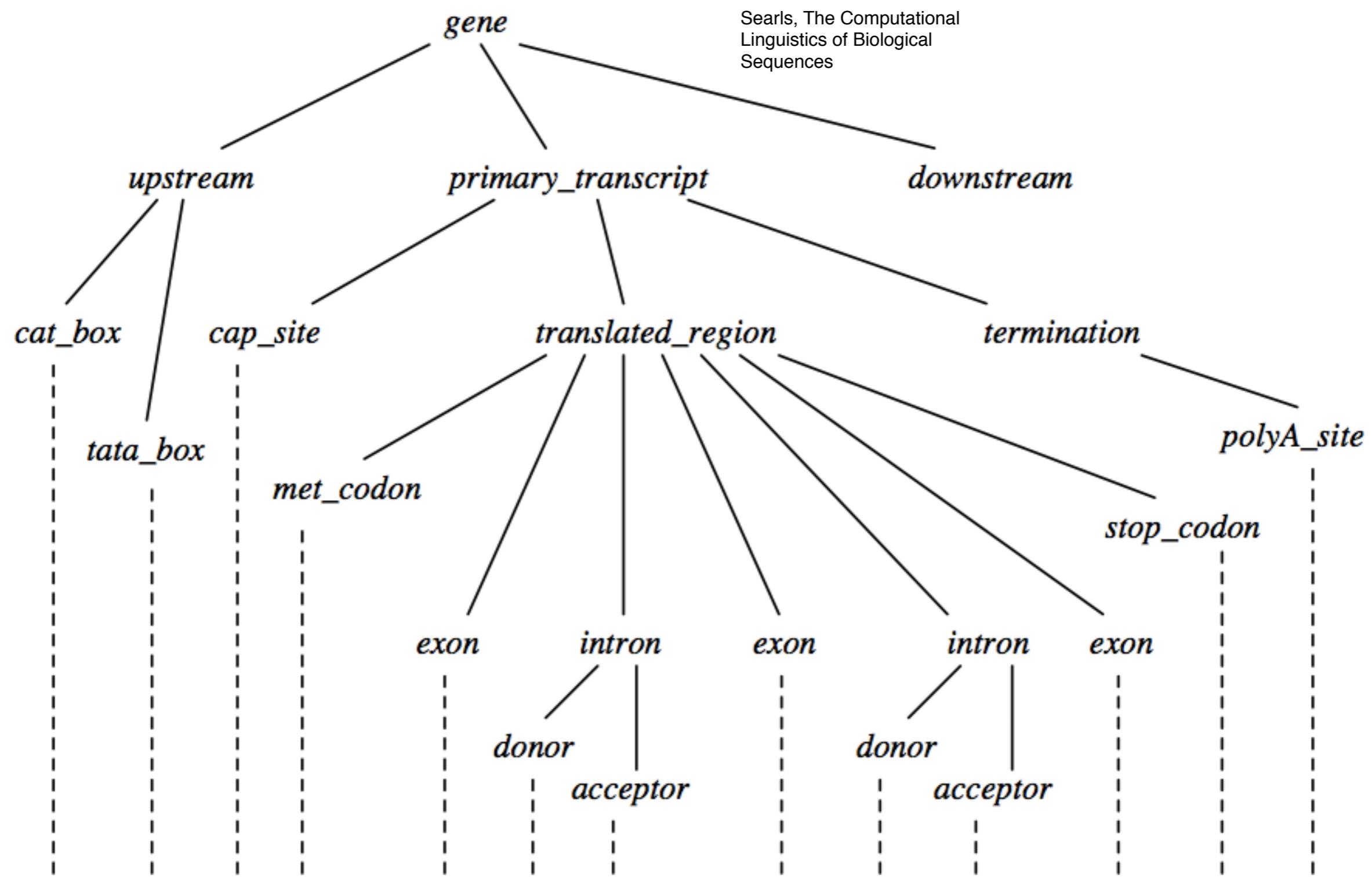


Eukaryotic genes usually look like this:



This spliced RNA is what is
translated into a protein.

Hypothetical Eukaryotic Gene Parse Tree



A human gene

chr11:5246500-5248500 (reverse strand):

```
ATATCTTAGAGGGAGGGCTGAGGGTTGAAGTCCAACCTCTAACGCCAGTGCAGAAGAGCCAAGGACAGGTACGGCTGTC  
ATCACCTAGACCTCACCCGTGGAGCCACACCCCTAGGGTGGCCAATCTACTCCCAGGAGCAGGGAGGGCAGGAGCCAGG  
GCTGGGCATAAAAGTCAGGGCAGAGCCATCTATTGCTTACATTGCTTCTGACACAACGTGTTCACTAGCAACCTCAAA  
CAGACACCATGGTGCATCTGACTCCTGAGGAGAAGTCTGCCGTTACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAGTT  
GGTGGTGAGGCCCTGGCAGGTTGGTATCAAGGTTACAAGACAGGTTAACGGAGACCAATAGAAACTGGCATGTGGAGA  
CAGAGAAGACTCTGGGTTCTGATAGGCAGTGCACCTCTGCCTATTGGTCTATTCCCACCCCTAGGCTGCTGGTG  
GTCTACCCTGGACCCAGAGGTTCTTGAGTCCTTGGGATCTGTCCACTCCTGATGCTGTTATGGCAACCTAAGGT  
GAAGGCTCATGGCAAGAAAGTGCTCGGTGCCTTAGTGTGGCCTGGCTCACCTGGACAACCTCAAGGGCACCTTCCA  
CACTGAGTGAGCTGACTGTGACAAGCTGCACGTGGATCCTGAGAACCTCAGGGTGAGTCTATGGACGCTGATGTTT  
CTTCCCCTTTCTATGGTTAACGTTATGTCATAGGAAGGGATAAGTAACAGGGTACAGTTAGAATGGAAAACAG  
ACGAATGATTGCATCAGTGTGGAAAGTCTCAGGATCGTTAGTTCTTTATTGCTGTTATAACAATTGTTTCTTT  
GTTAATTCTGCTTCTTTCTCCGCAATTACTATTACTTAATGCCTAACATTGTGTATAACAAA  
AGGAAATATCTCTGAGATACATTAAGTAACCTAAAAAAACTTACACAGTCTGCCTAGTACATTACTATTGGAATAT  
ATGTGTGCTTATTGCATATTATAATCTCCCTACTTATTCTTTATTGATAACATAATTACATAT  
TTATGGGTTAAAGTGTAAATGTTAATATGTGTACACATATTGACCAAATCAGGGTAATTGCTATTGTAATTAAAA  
AATGCTTCTTCTTTAATATACTTTGTTATCTTAACTTCCAACTCTCTTCAATTGCTTCAATCTCTTCTTCAAGGGCAATAA  
TGATACAATGTATCATGCCTCTTGACCACTAAAGAATAACAGTGATAATTCTGGGTTAAGGCAATAGCAATATCT  
CTGCATATAAAATTCTGCATATAAAATTGTAACGTGATGTAAGAGGTTCATATTGCTAATAGCAGCTACAATCCAGCTA  
CCATTCTGCTTTATTGTTGGGATAAGGCTGGATTATTCTGAGTCCAAGCTAGGCCCTTGCTAATCATGTTCA  
TACCTCTATCTCCTCCCACAGCTCCTGGCAACGTGCTGGTCTGTGCTGGCCCATCACTTGGCAAAGAATTCA  
CCACCAAGTGCAGGCTGCCTATCAGAAAGTGGTGGCTGGCTAATGCCCTGGCCCACAAGTATCAAGCTCGCTT  
TCTTGCTGCTCAATTCTATTAAAGGTTCCCTTGCTTAAGTCCAACACTAAACTGGGGATATTATGAAGGGCCTT  
GAGCATCTGGATTCTGCCTAATAAAACATTATTGCAATGATGTATTAAATTATTCTGAATATTACTA  
AAAAGGAAATGTGGAGGTAGTGCATTAAACATAAAAGAAATGAAGAGCTAGTTCAAACCTTGGAAAATACACTATA  
TCTTAAACTCCATGAAAGAAGGTGAGGCTGCAAACAGCTAATGCACATTGGCAACAGCCCCTGATGCATATGCCTTATT
```

A human gene

chr11:5246500-5248500 (reverse strand):

ATATCTTAGAGGGAGGGCTGAGGGTTGAAGTCCAACCTCCTAACGCCAGTGCCAGAAGAGCCAAGGCACAGGTACGGCTGTC
ATCACTTAGACCTCACCTGTGGAGCCACACCCTAGGGTTGCCAATCTACTCCCAGGAGCAGGGAGGGCAGGAGCCAGG
GCTGGGCATAAAAGTCAGGGCAGAGCCATCTATTGCTTACATTGCTTCTGACACAACGTGTTCACTAGCAACCTCAAA
CAGACACC**ATGGTGCATCTGACTCCTGAGGAGAAGTCTGCCGTTACTGCCCTGTGGGGCAAGGTGAACGTGGATGAAGTT**
GGTGGTGAGGCCCTGGCAGGTTGGTATCAAGGTTACAAGACAGGTTAAGGAGACCAATAGAAACTGGCATGTGGAGA
CAGAGAAGACTCTTGGGTTCTGATAGGCAGTGCCTCTGCCTATTGGCTATTTCACCCCTAG**GCTGCTGGTG**
GTCTACCCTGGACCCAGAGGTTCTTGAGTCCTTGGGGATCTGCCACTCCTGATGCTGTTATGGCAACCTAAGGT
GAAGGCTCATGGCAAGAAAGTGCTCGGTGCCCTTAGTGATGCCCTGGCTCACCTGGACAACCTCAAGGGCACCTTGC
CACTGAGTGAGCTGCACTGTGACAAGCTGACGTGGATCCTGAGAACTTCAGGGTGAGTCTATGGACGCTTGATGTTT
CTTCCCCCTTTCTATGGTTAAGTTATGTCATAGGAAGGGATAAGTAACAGGGTACAGTTAGAATGGAAACAG
ACGAATGATTGCATCAGTGTGGAAGTCTCAGGATCGTTAGTTCTTTATTGCTGTTCATACAAATTGTTCTTT
GTTAATTCTGCTTCTTTCTTCTCCGCAATTAACTATTAACTTAATGCCTAACATTGTGTATAACAAA
AGGAAATATCTCTGAGATACATTAAAGTAACAAAAAAACTTACACAGTCTGCCTAGTACATTACTATTGGAATAT
ATGTGTGCTTATTGCAT**Homo sapiens hemoglobin, beta (HBB)**
TTATGGGTTAAAGTGTAA
AATGCTTCTTAAATATACTTTGTTATCTTAACTTCTAACATTCCCTAACATTCTTTCTTCAGGGCAATAA
TGATACAATGTATCATGCCTCTTGACCAATTCTAAAGAATAACAGTGATAATTCTGGGTTAAGGCAATAGCAATATCT
CTGCATATAAAATTCTGCATATAAAATTGTAACGTGATGTAAGAGGTTCATATTGCTAACAGCTACATCCAGCTA
CCATTCTGCTTTATTGTTGGGATAAGGCTGGATTATTCTGAGTCCAAGCTAGGCCCTTGCTAACATGTTCA
TACCTCTTATCTCCTCCCACAG**CTCCTGGCAACGTGCTGGCTGTGCTGCCCATCACTTGGCAAAGAATT**
CACCAAGTGCAGGCTGCCTATCAGAAAGTGGTGGCTGGCTGGCTAACATGCCCTGGCCCACAAGTATCACTAAGCTCGCTT
TCTTGCTGTCCAATTCTATTAAAGGTTCTTGTCCCTAACGTCCAACACTAAACTGGGGATATTATGAAGGGCCTT
GAGCATCTGGATTCTGCCTAACAAAAACATTATTGCAATGATGTATTAAATTATTCTGAATATTACTA
AAAAGGGAATGTGGGAGGTCACTGCATTAAAACATAAGAAATGAAGAGCTAGTTCAAACCTTGGAAAATACACTATA
TCTTAAACTCCATGAAAGAAGGTGAGGCTGCAAACAGCTAACATTGGCAACAGCCCCTGATGCATATGCCTTATT

Advancing the State-of-the-art in Computational Gene Prediction

Bill Majoros (bmajoros@duke.edu)



Following slides (marked #) are from: <http://www.geneprediction.org/tutorial-majoros.pdf>

Phase Constraints

phase: 012012012012012012
sequence: + **ATGCGATA****TGA****TCGC****TAG**
coordinates: 0 5 10 15

forward strand

phase: 210210210210210210
sequence: + **CTAGCGAT****CATATCGC****CAT**
- **GATCGCTAGT****ATAGCGTA**
coordinates: 0 5 10 15

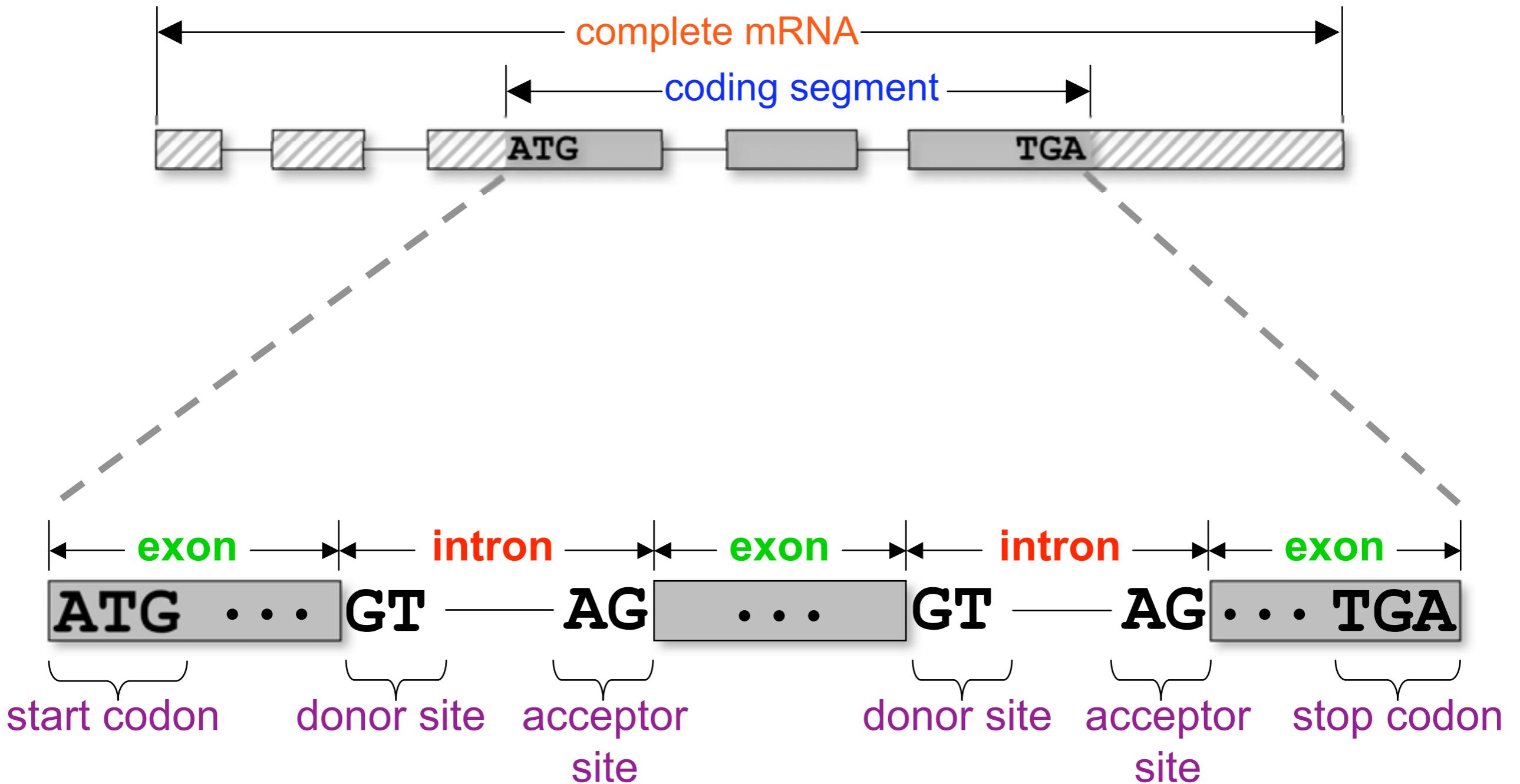
reverse strand

phase: 01201201 2012012012
sequence: + G**TATGCGATA**GTCAAGAG**TGATCGCTAG**ACC
coordinates: 0 5 10 15 20 25 30

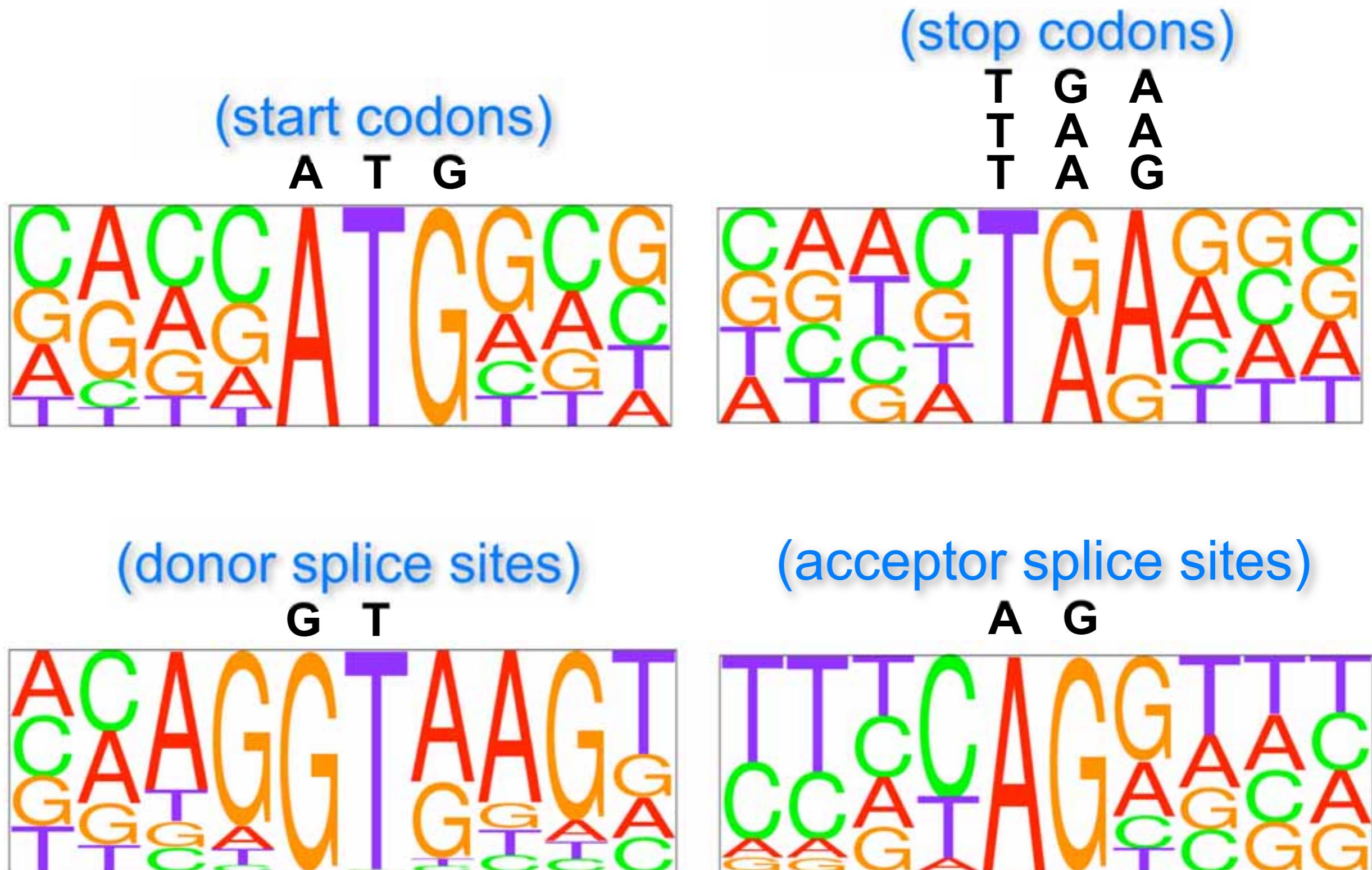
forward strand,
spliced



The Eukaryotic Gene Finding Problem



The Stochastic Nature of Signal Sensing



Common Assumptions in Gene Finding

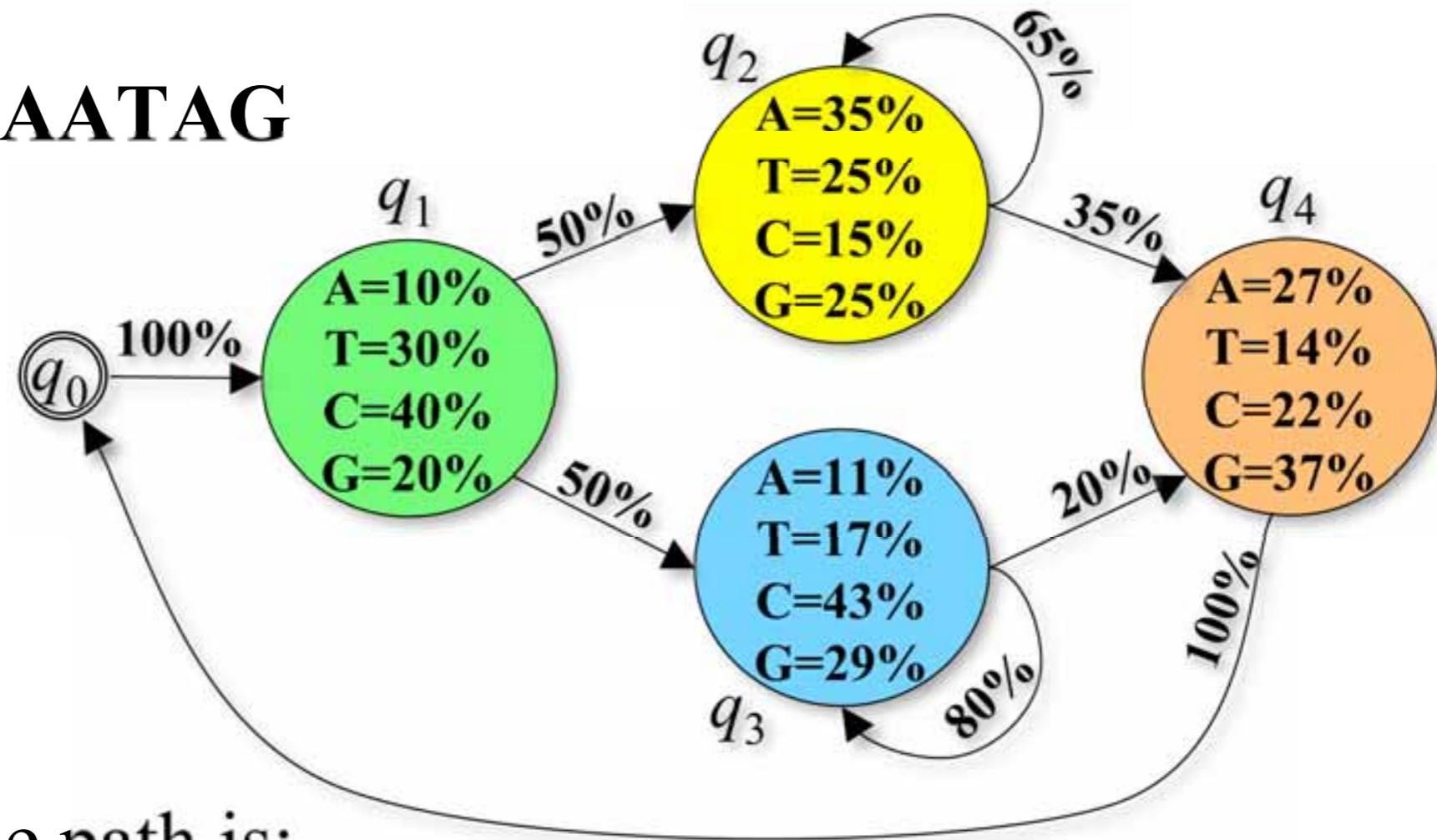
- No overlapping genes
- No nested genes
- No frame shifts or sequencing errors
- Optimal parse only
 - No split start codons (**ATGT...AGG**)
 - No split stop codons (**TGT...AGAG**)
- No alternative splicing
- No selenocysteine codons (TGA)
- No ambiguity codes (Y,R,N, etc.)

Finding the Most Probable Path

Example: CATTAAATAG

top: 7.0×10^{-7}

bottom: 2.8×10^{-9}



The most probable path is:

States: **122222224** ↗
Sequence: CATTAAATAG

resulting in this parse:

States: 122222224 ↗
Sequence: **CATTAAATAG**

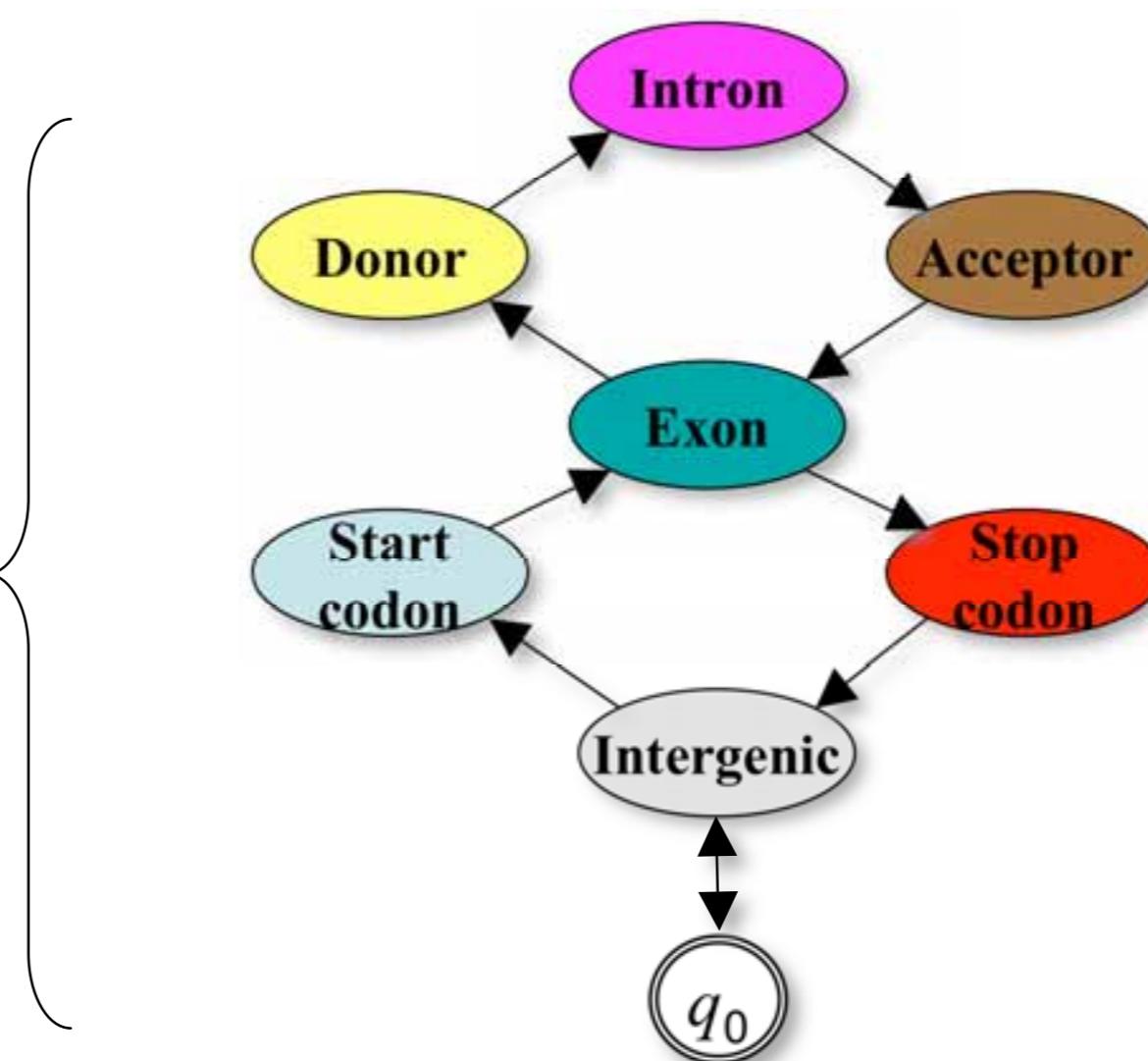
feature 1: **C**

feature 2: **ATTAATA**

feature 3: **G**

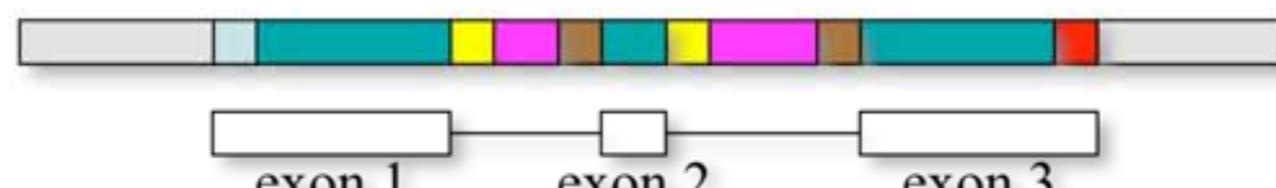
Using an HMM for Gene Prediction

the Markov model:

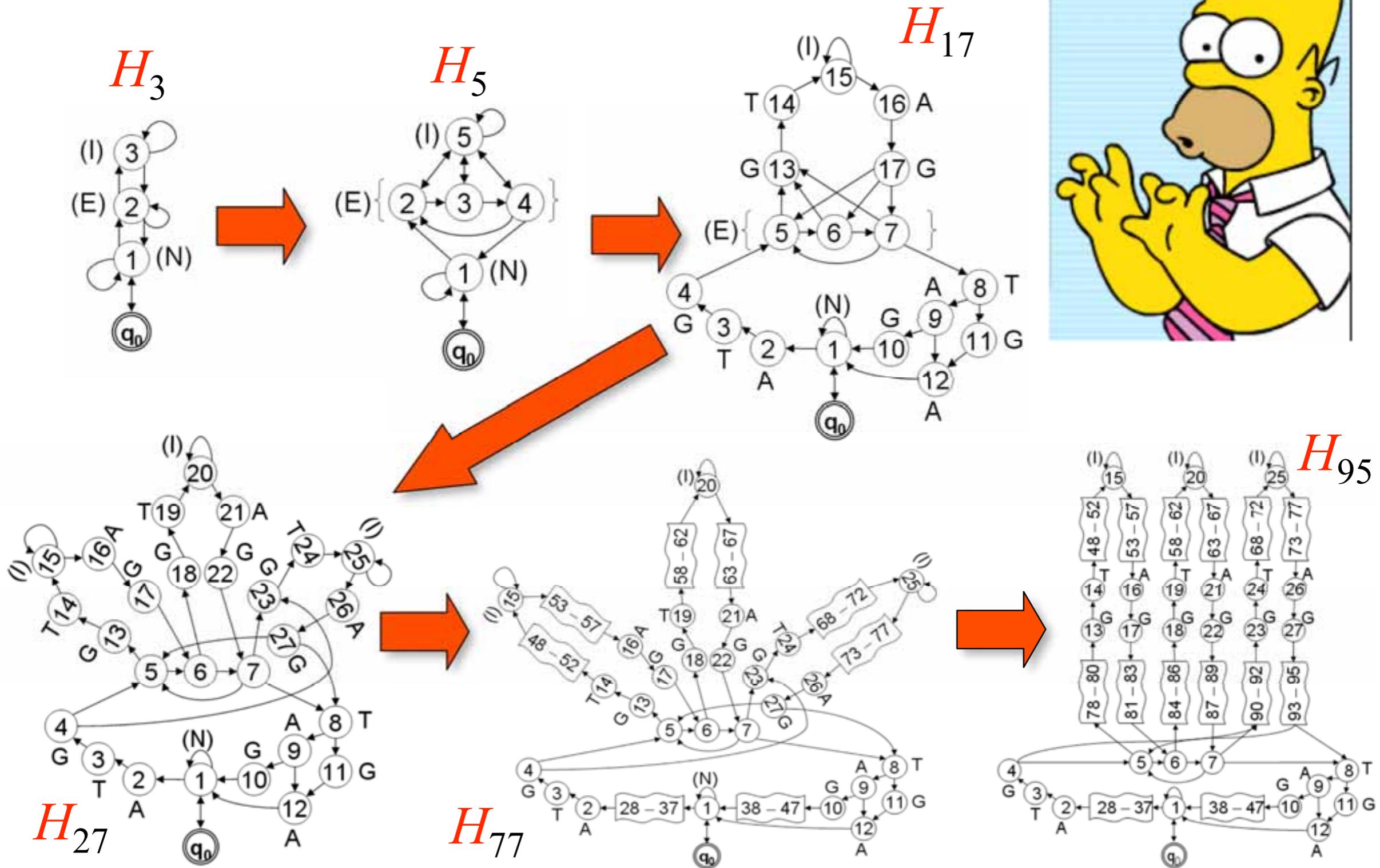


the input sequence:
the most probable path:
the gene prediction:

AGCTAGCAGTATGTCATGGCATGTTGGAGGTAGTACGTAGAGGTAGCTAGTATAGGTCGATAGTACCGGA

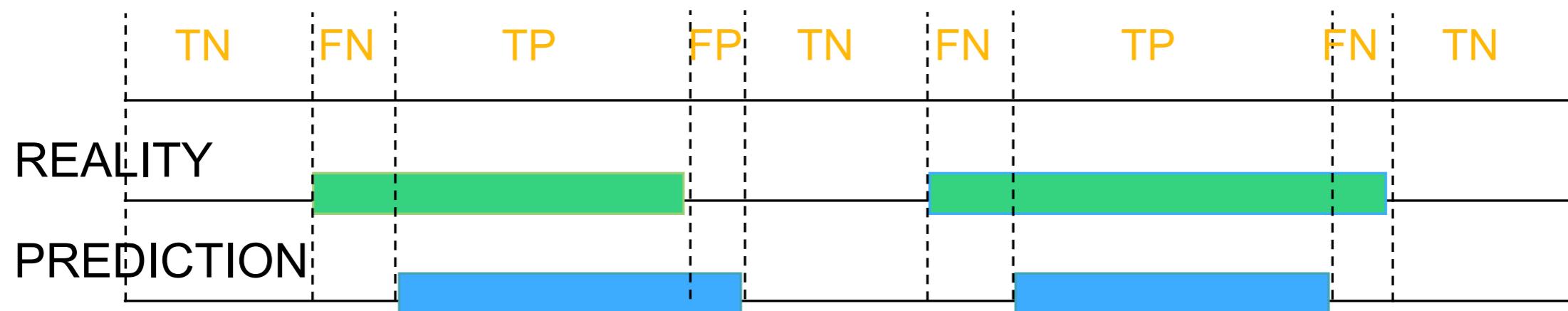


Higher Order Markovian Eukaryotic Recognizer (HOMER)



Evaluation of Gene Finding Programs

Nucleotide level accuracy



Sensitivity:

$$Sn = \frac{TP}{TP + FN}$$

What fraction of reality did you predict?

Specificity:

$$Sp = \frac{TP}{TP + FP}$$

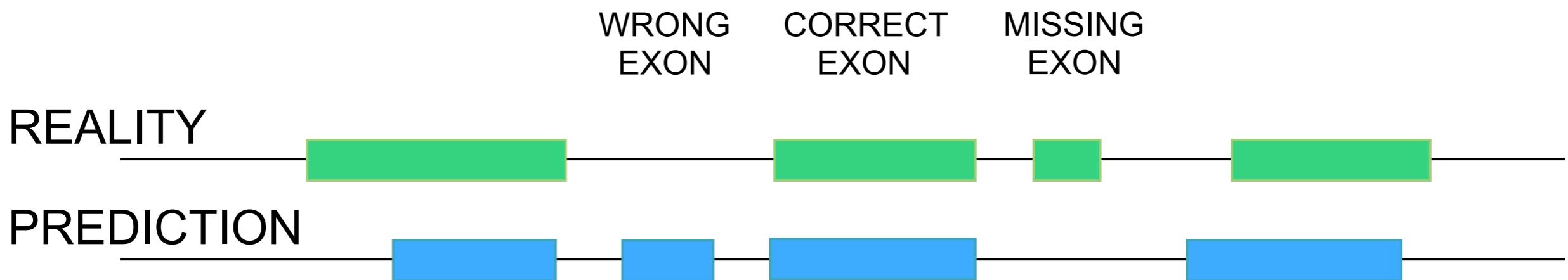
What fraction of your predictions are real?

(actually, this is Precision,
which is not the same thing)

Nonetheless, this measure is used below and is called "Specificity" in the gene-finding community. For more details, see Burset, Moises, and Roderic Guigo. "Evaluation of gene structure prediction programs." Genomics 34.3 (1996): 353-367.

More Measures of Prediction Accuracy

Exon level accuracy



$$ExonSn = \frac{TE}{AE} = \frac{\text{number of correct exons}}{\text{number of actual exons}}$$

$$ExonSp = \frac{TE}{PE} = \frac{\text{number of correct exons}}{\text{number of predicted exons}}$$

Defined equiv. for splice sites & start/stop codons

HOMER, version H_3



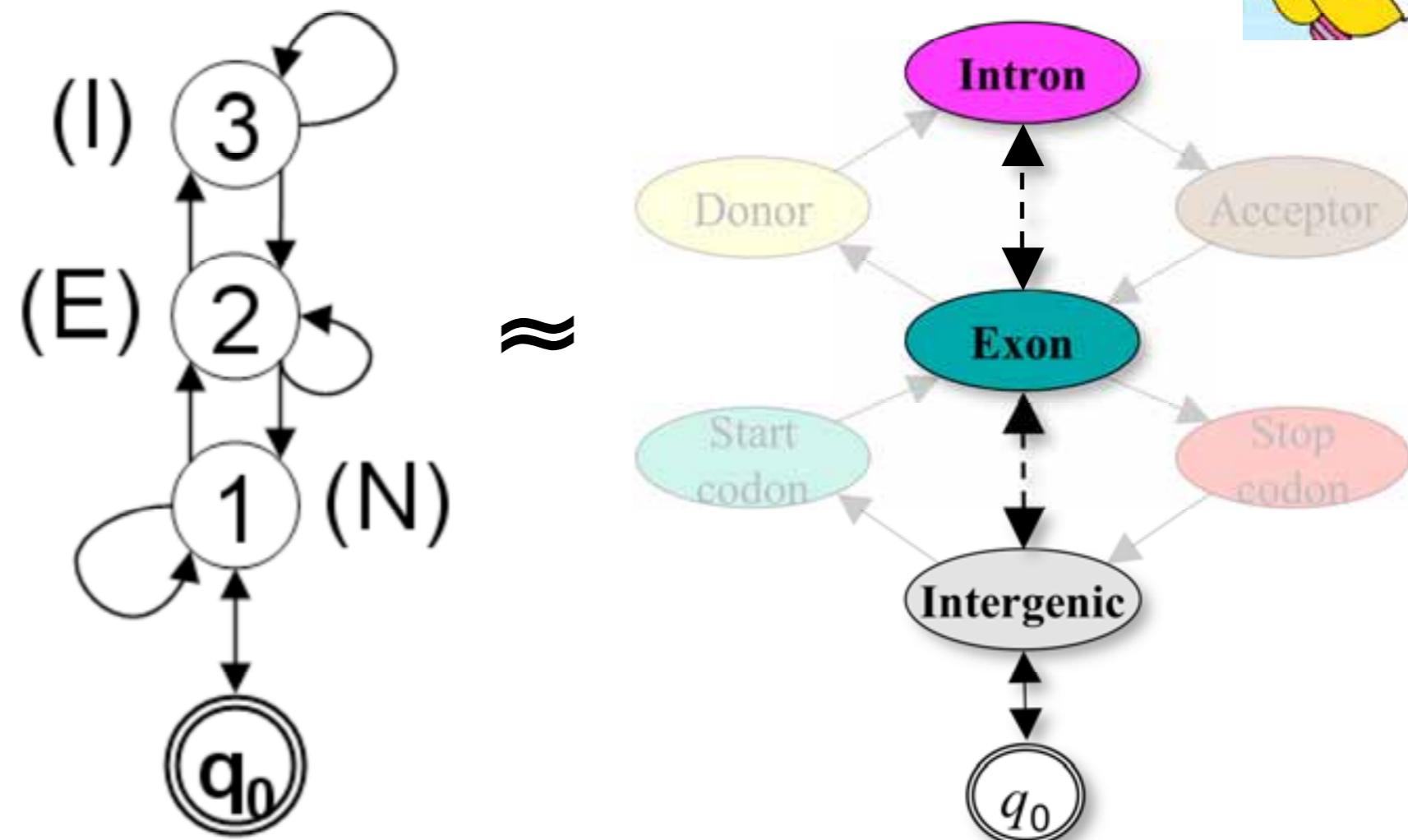
I=intron state

E=exon state

N=intergenic state

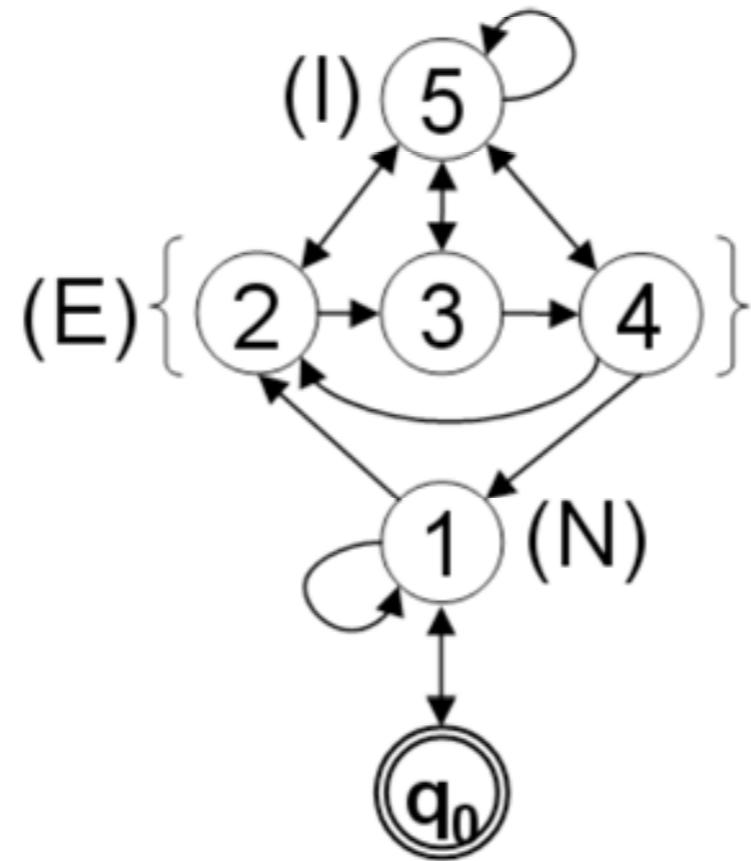
tested on 500

Arabidopsis genes:



	nucleotides			splice sites		start/stop codons		exons			genes	
	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	#
baseline	50	28	44	0	0	0	0	0	0	0	0	0
H_3	53	88	66	0	0	0	0	0	0	0	0	0

HOMER, version H_5

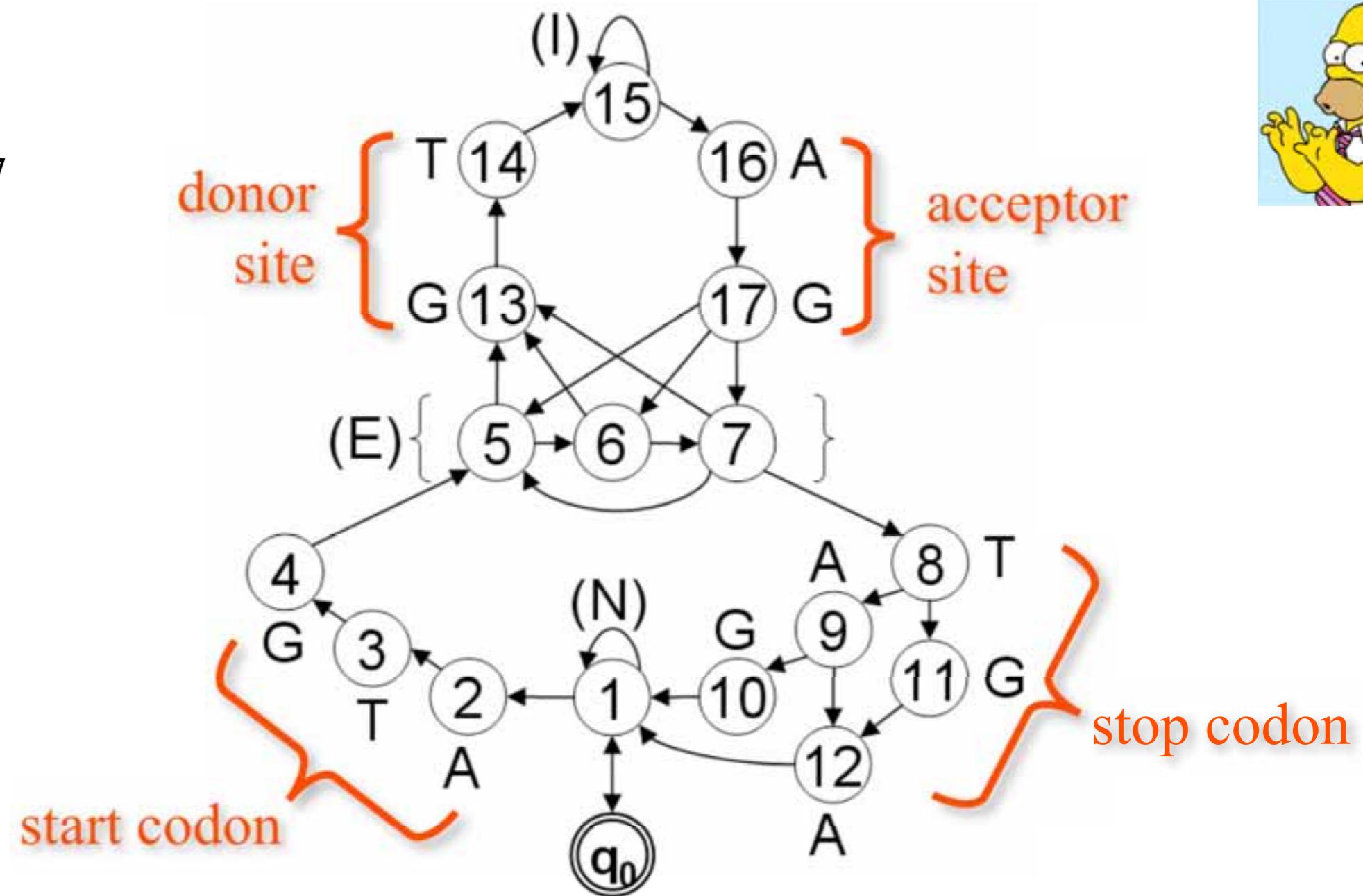


three exon states, for
the three codon
positions

	nucleotides			splice sites		start/stop codons		exons			genes	
	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	#
H_3	53	88	66	0	0	0	0	0	0	0	0	0
H_5	65	91	76	1	3	3	3	0	0	0	0	0

HOMER

version H_{17}

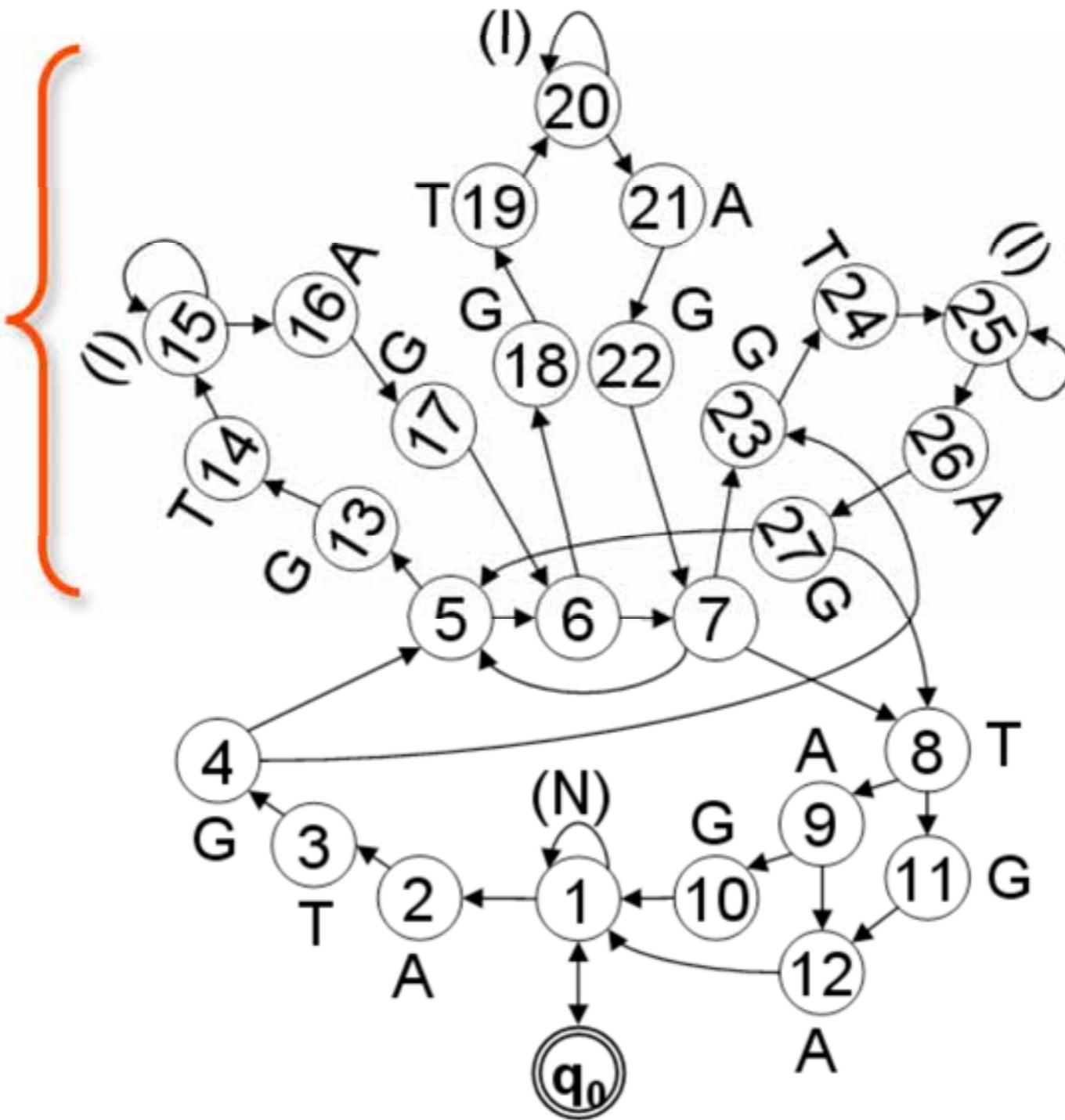


	nucleotides			splice sites		start/stop codons		exons			genes	
	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	#
H_5	65	91	76	1	3	3	3	0	0	0	0	0
H_{17}	81	93	87	34	48	43	37	19	24	21	7	35

HOMER

version H_{27}

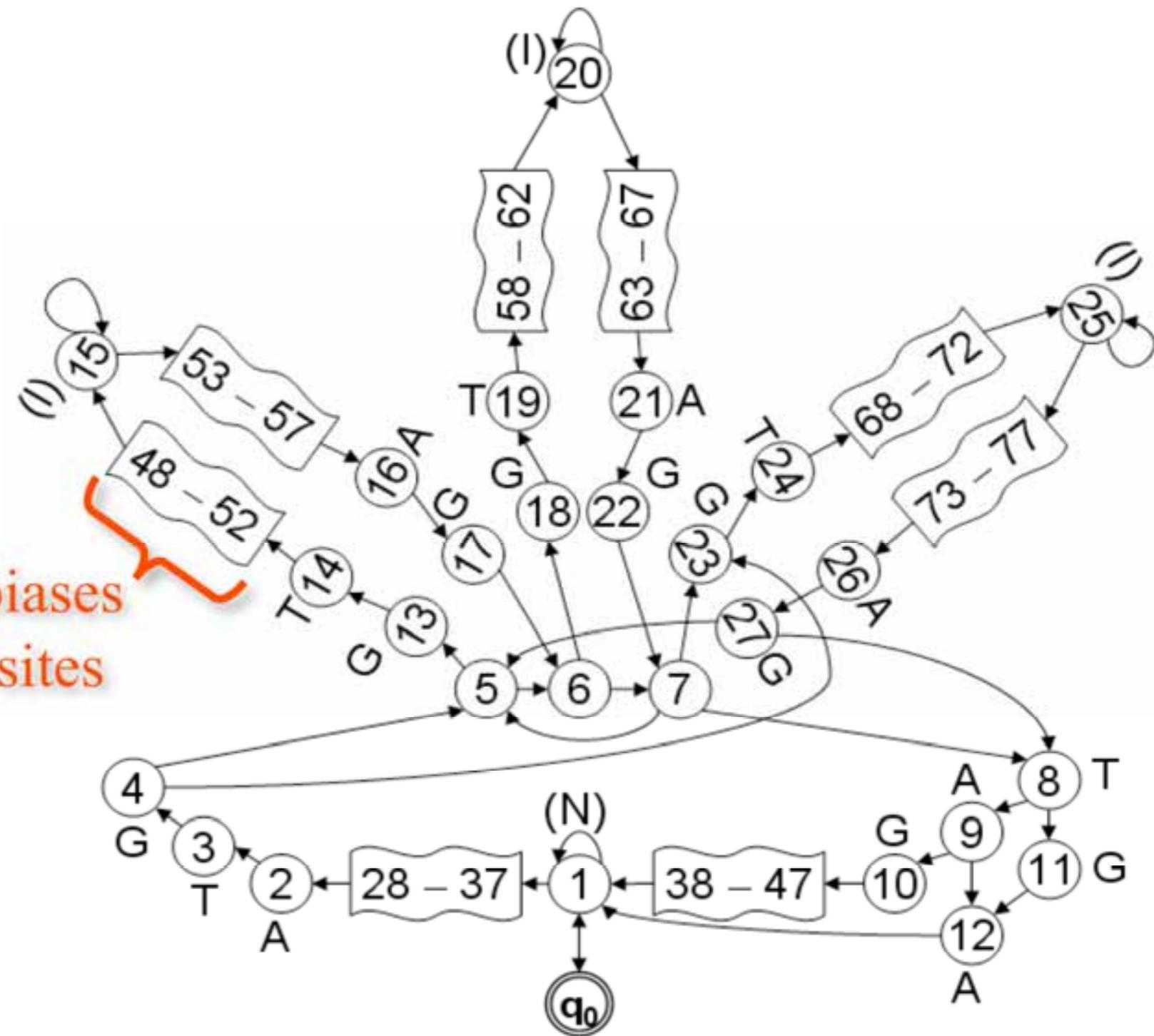
three
separate
intron
models



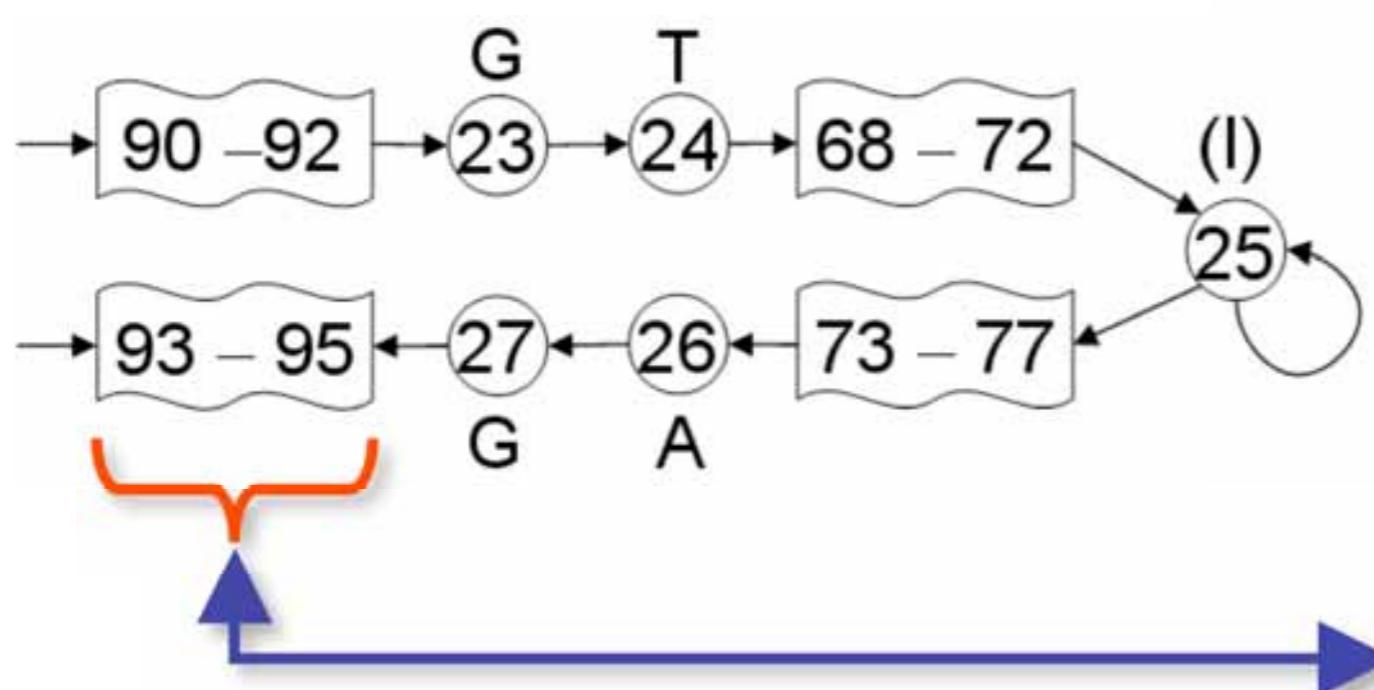
	nucleotides			splice		start/stop		exons			genes	
	Sn	Sp	F	Sn	Sp	Sn	Sp	Sn	Sp	F	Sn	#
H_{17}	81	93	87	34	48	43	37	19	24	21	7	35
H_{27}	83	93	88	40	49	41	36	23	27	25	8	38

HOMER

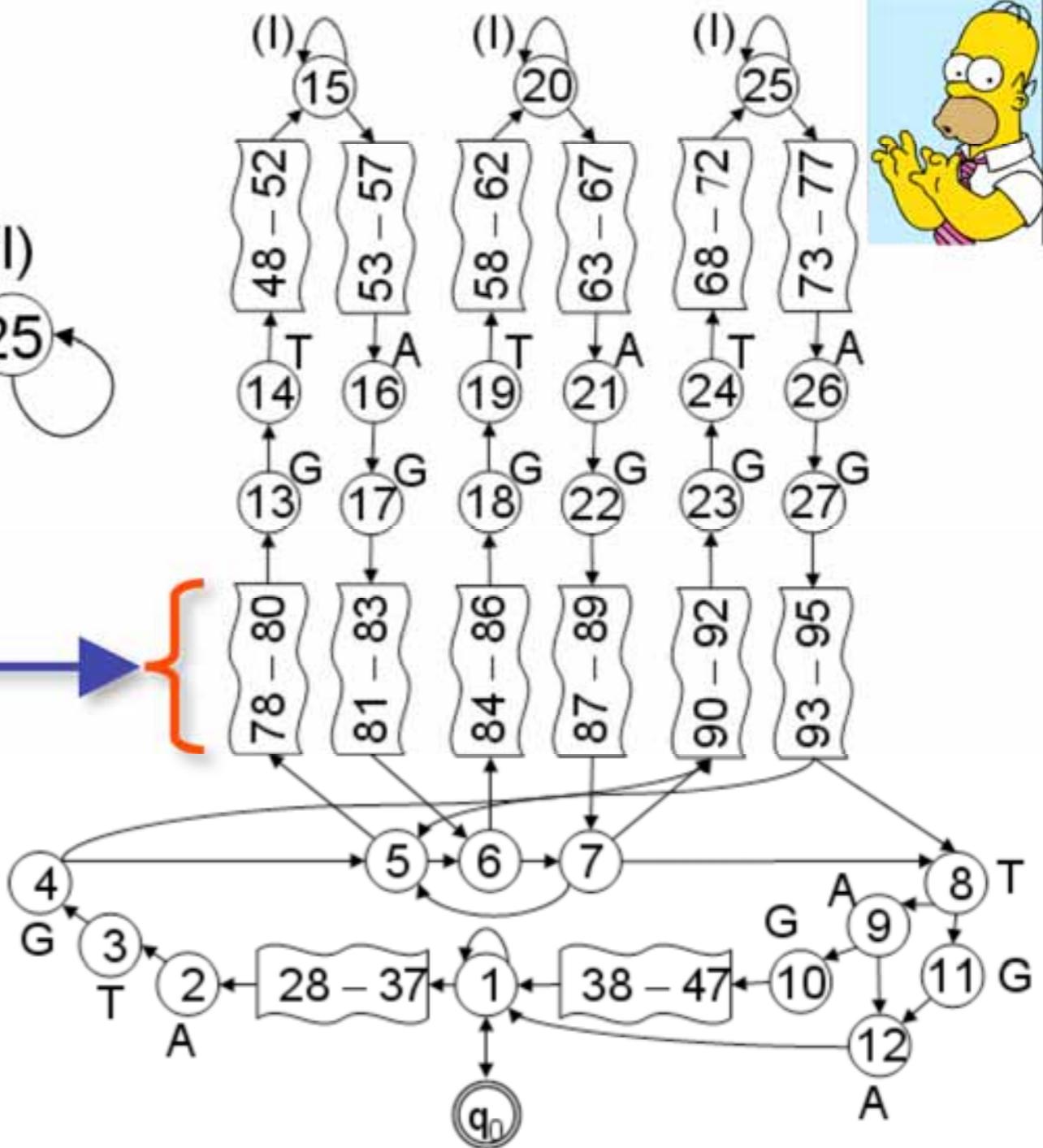
version H_{77}



	nucleotides			splice		start/stop		exons			genes	
	Sn	Sp	F	Sn	Sp	Sn	Sp	Sn	Sp	F	Sn	#
H_{27}	83	93	88	40	49	41	36	23	27	25	8	38
H_{77}	88	96	92	66	67	51	46	47	46	46	13	65



HOMER
version H_{95}



	nucleotides			splice		start/stop		exons			genes	
	Sn	Sp	F	Sn	Sp	Sn	Sp	Sn	Sp	F	Sn	#
H_{77}	88	96	92	66	67	51	46	47	46	46	13	65
H_{95}	92	97	94	79	76	57	53	62	59	60	19	93

Higher-order Markov Models

0th order:

A C **G** C T A

P(G)

1st order:

A C **G** C T A

P(G|C)

2nd order:

A C **G** C T A

P(G|AC)

Higher-order Markov Models

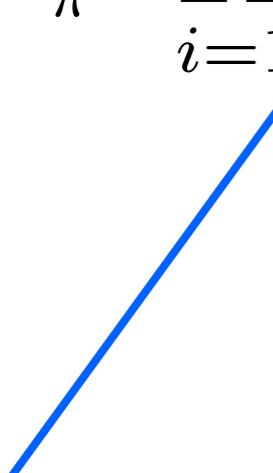
	order	nucleotides			splice sites		starts/stops		exons			genes	
		Sn	Sp	F	Sn	Sp	Sn	Sp	Sn	Sp	F	Sn	#
H_{95}^0	0	92	97	94	79	76	57	53	62	59	60	19	93
H_{95}^1	1	95	98	97	87	81	64	61	72	68	70	25	127
H_{95}^2	2	98	98	98	91	82	65	62	76	69	72	27	136
H_{95}^3	3	98	98	98	91	82	67	63	76	69	72	28	140
H_{95}^4	4	98	97	98	90	81	69	64	76	68	72	29	143
H_{95}^5	5	98	97	98	90	81	66	62	74	67	70	27	137

Generalized HMMs

- Each state can emit a sequence of symbols.
- In the diagrams on the next few slides, each state emits a *complete gene feature* (e.g. an entire exon):

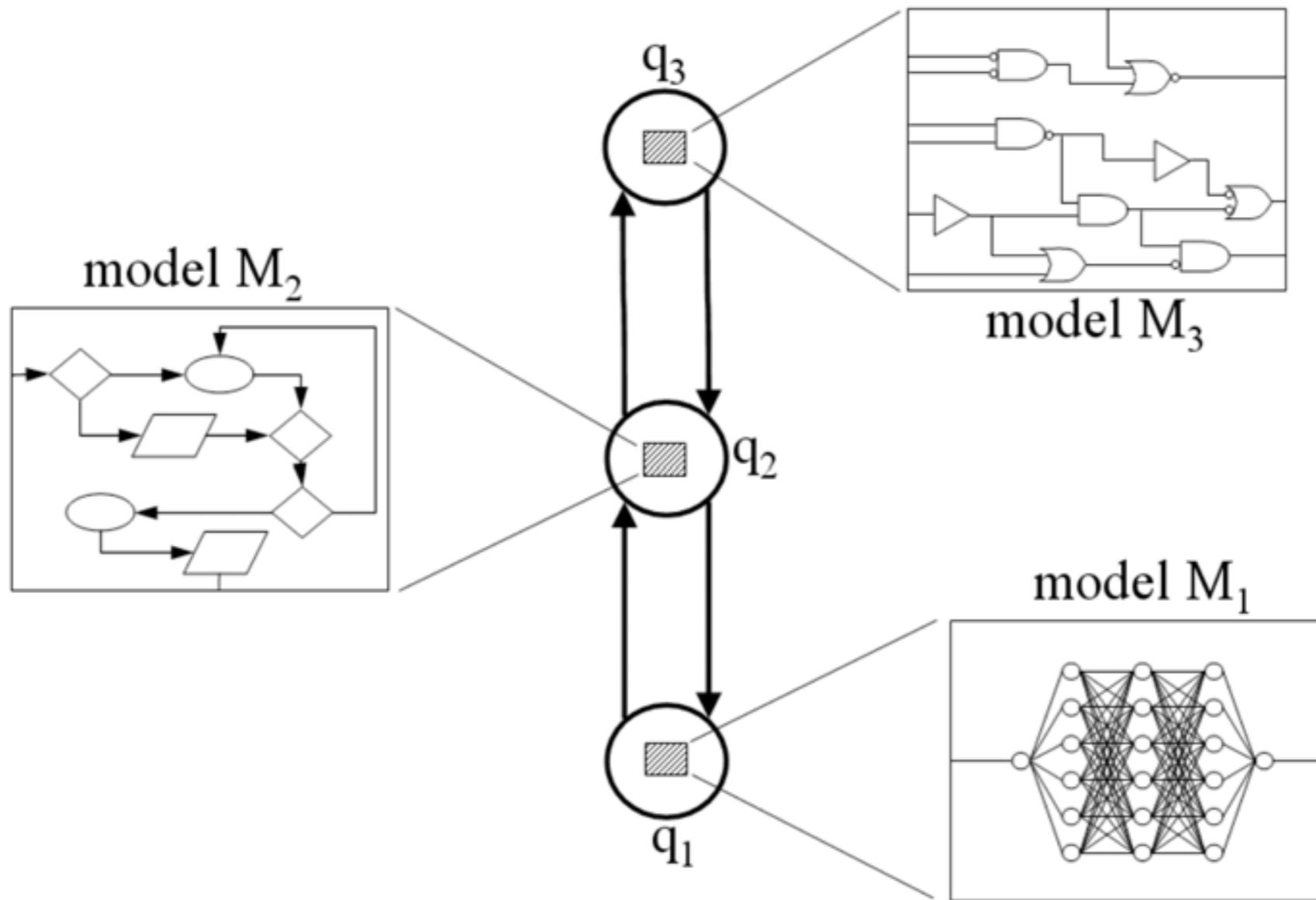
$$\max_{\pi} \prod_{i=1}^n \Pr(x_i \dots x_{i+d_i} \mid \pi_i, d_i) \underbrace{\Pr(d_i \mid \pi_i)}_{\text{Probability that the state will emit } d_i \text{ symbols.}} \underbrace{\Pr(\pi_i \rightarrow \pi_{i+1})}_{\text{Probability of transitioning to the next state}}$$

Probability of emitting the string of length d_i .



This probability could itself be computed by an HMM or a Markov chain, etc.

Generalized Hidden Markov Models



Advantages:

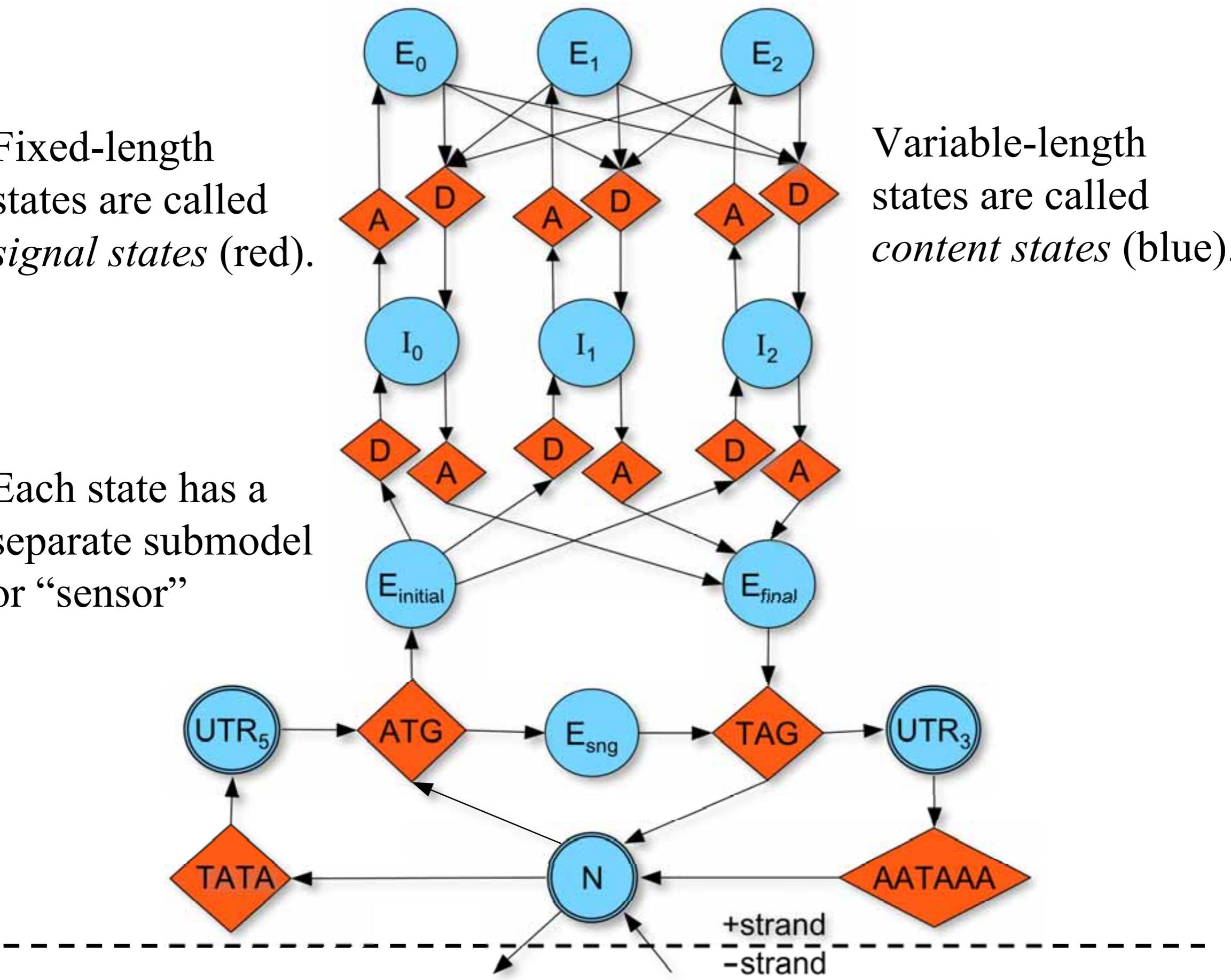
- * Submodel abstraction
- * Architectural simplicity
- * State duration modeling

Disadvantages:

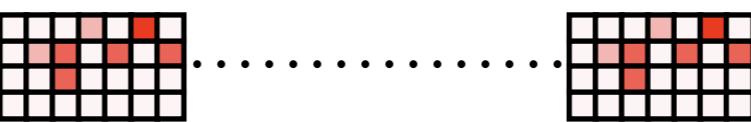
- * Decoding complexity

Fixed-length
states are called
signal states (red).

Each state has a
separate submodel
or “sensor”



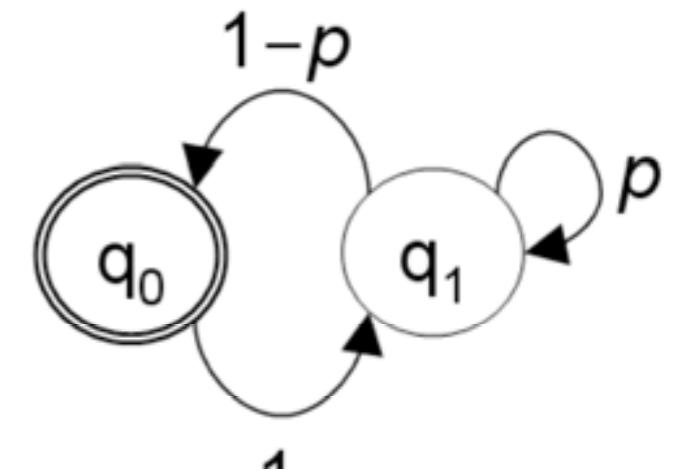
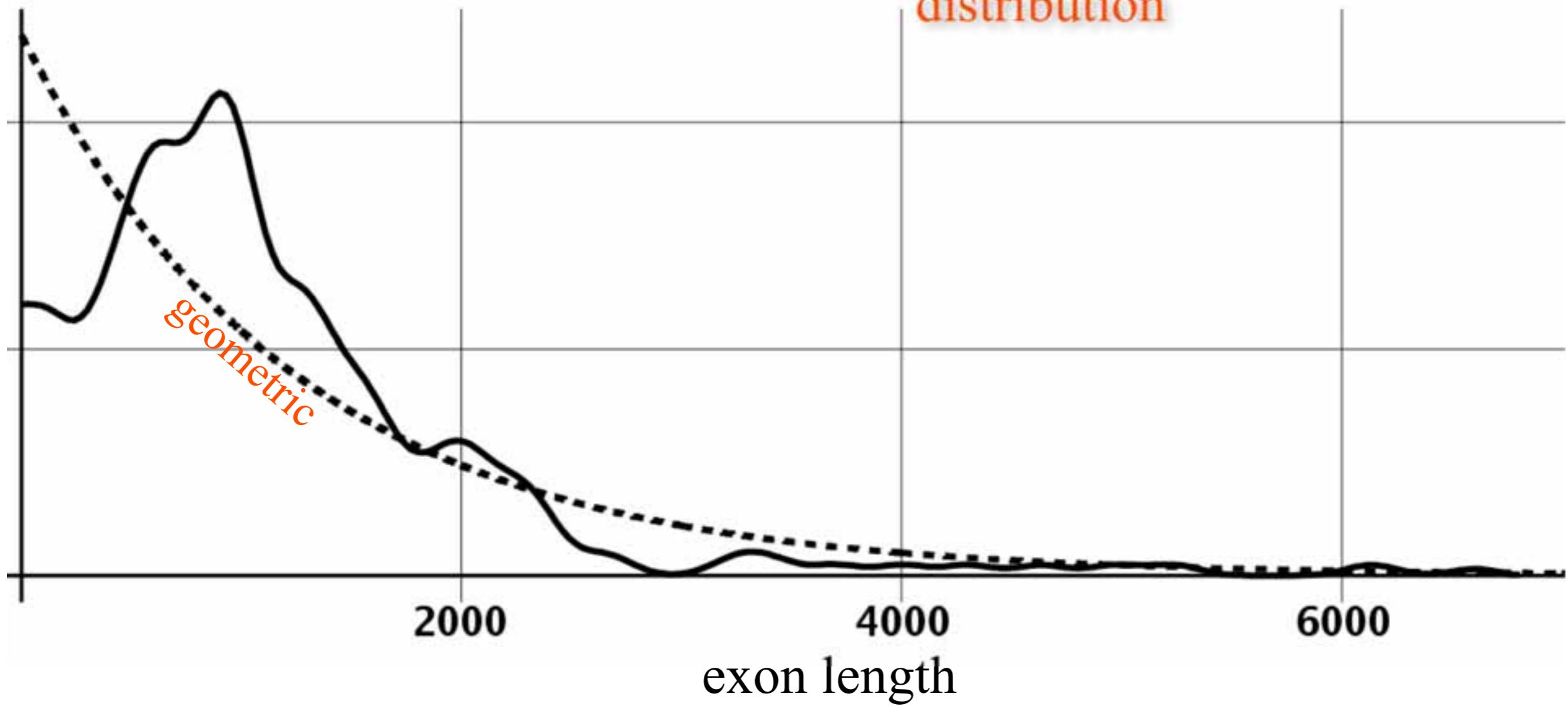
Components Needed

- Probability distribution of initial state
 - = the fraction of known genome corresponding to each state, divided into groups by GC content.
- State transition probabilities
 - = the probability X follows Y in known genes
- Length distributions for each state
 - For exons: = estimated from empirically observed distribution
 - For introns: = geometric distribution with parameter q_{gc} , where is the best fit parameter for regions with a given GC content.
- Sequence models for each state/length
 - for states with strong motifs:

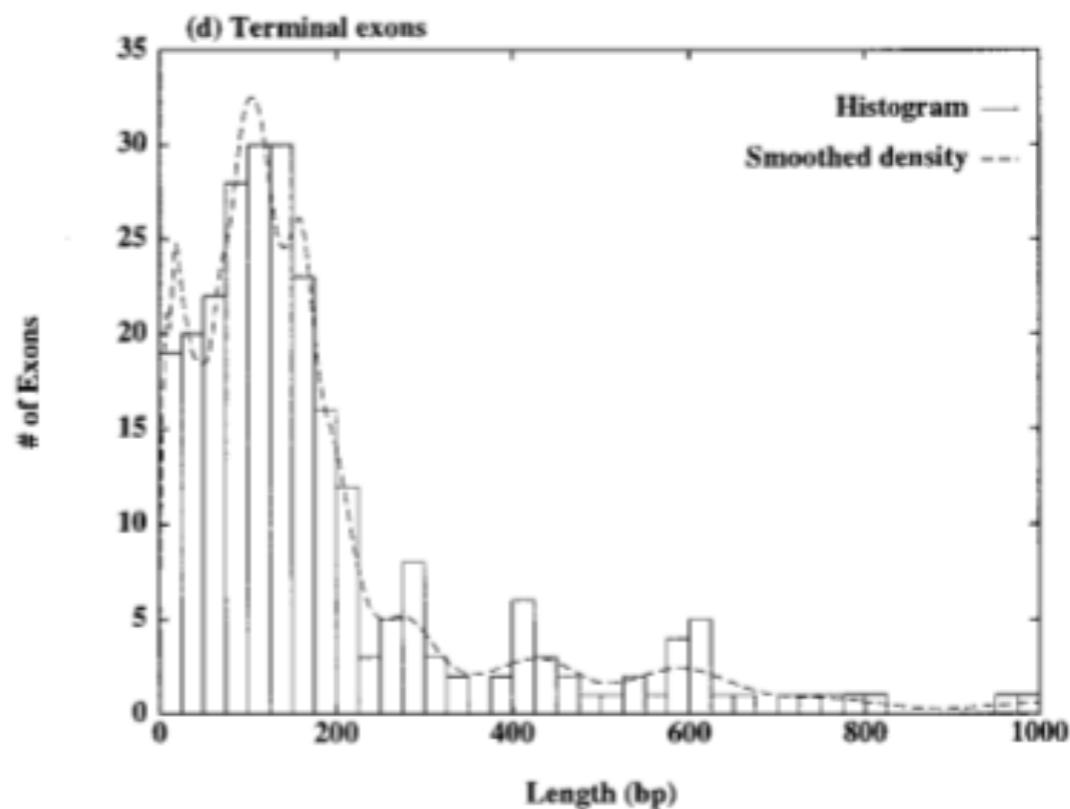
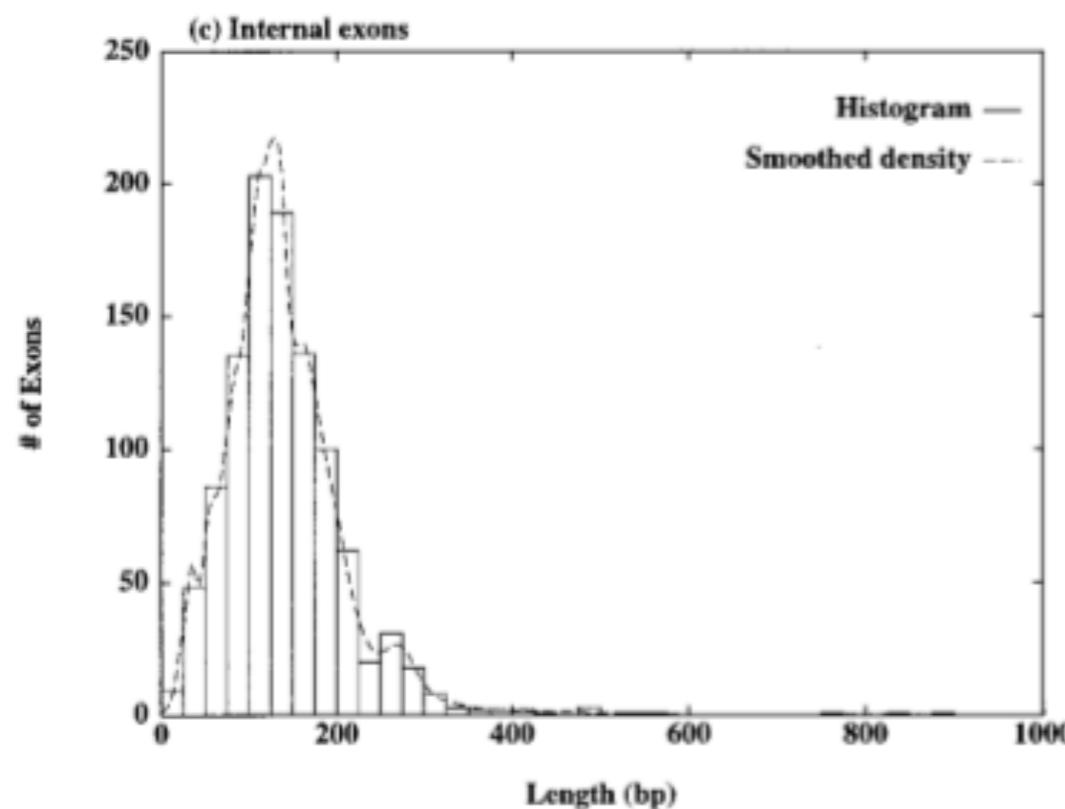
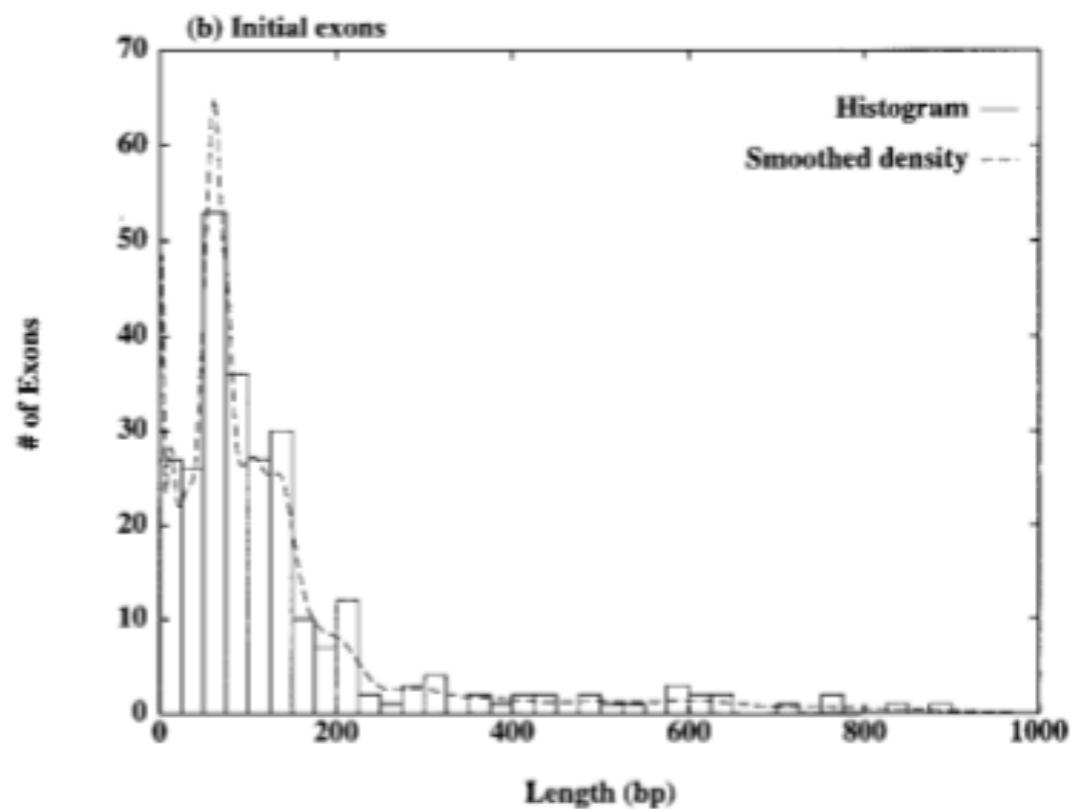
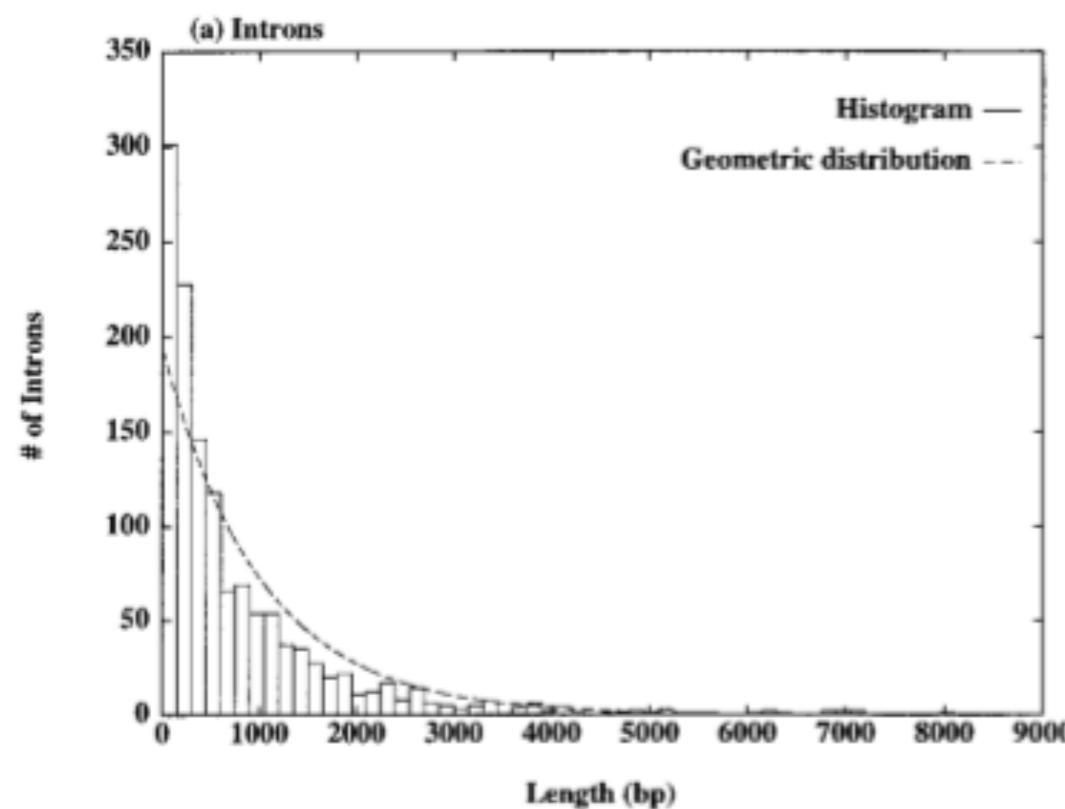
HMMs & Geometric Feature Lengths

$$P(x_0 \dots x_{d-1} | \theta) = \left(\prod_{i=0}^{d-1} P_e(x_i | \theta) \right) p^{d-1} (1-p)$$

geometric
distribution



Feature Length Distributions

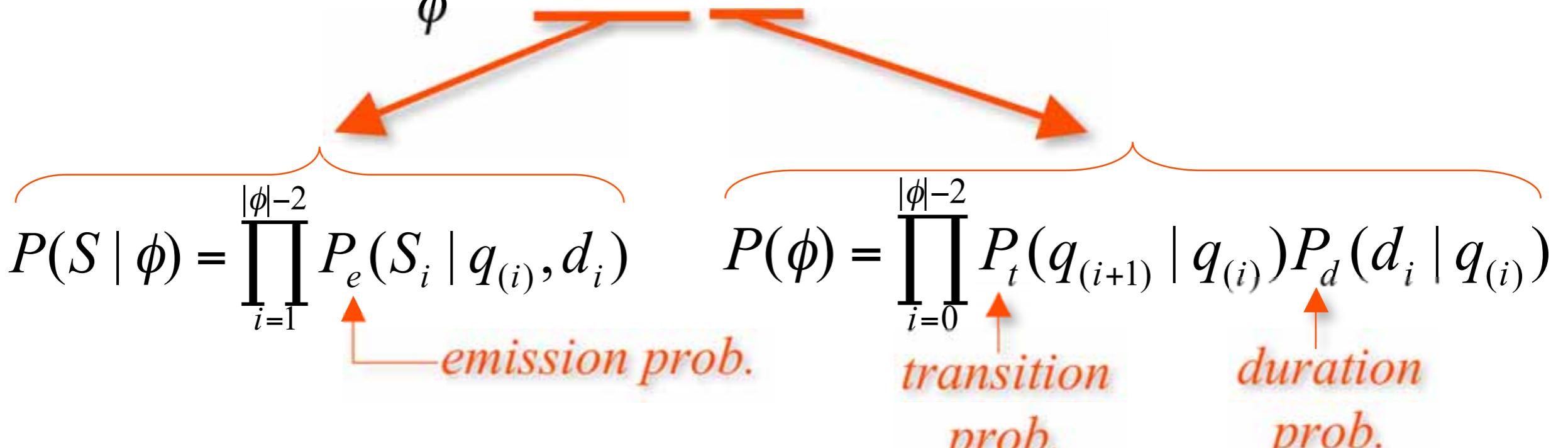


Decoding with a GHMM

$$\phi_{\max} = \underset{\phi}{\operatorname{argmax}} P(\phi | S) = \underset{\phi}{\operatorname{argmax}} \frac{P(\phi \wedge S)}{P(S)}$$

$$= \underset{\phi}{\operatorname{argmax}} P(\phi \wedge S)$$

$$= \underset{\phi}{\operatorname{argmax}} P(S | \phi)P(\phi)$$



$$\phi_{\max} = \underset{\phi}{\operatorname{argmax}} \prod_{i=0}^{|\phi|-2} P_e(S_i | q_{(i)}, d_i)P_t(q_{(i+1)} | q_{(i)})P_d(d_i | q_{(i)})$$

Accuracy

	nucleotides			splice		start/stop		exons			genes	
	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>Sn</i>	<i>Sp</i>	<i>F</i>	<i>Sn</i>	#
HMM	98	97	98	90	81	66	62	74	67	70	27	
GHMM	94	96	95	87	89	77	74	79	80	80	45	

GlimmerHMM (a generalized HMM for gene finding)

% of predicted in-gene nucleotides that are correct



% of predicted exons that are true exons.



	Nuc Sens	Nuc Prec	Nuc Accur	Exon Sens	Exon Prec	Exact Genes	Size of test set
D.rerio	93%	78%	86%	77%	69%	24%	549 genes
C.elegans	96%	95%	96%	82%	81%	42%	1886 genes
Arabidopsis	97%	99%	98%	84%	89%	60%	809 genes
Cryptococcus	96%	99%	98%	86%	88%	53%	350 genes
Coccidioides	99%	99%	99%	84%	86%	60%	503 genes
Brugia	93%	98%	95%	78%	83%	25%	477 genes

% of true gene nucleotides that GlimmerHMM predicts as part of genes.



% of true exons that GlimmerHMM found.



% of genes perfectly found



Compare with GENSCAN (another generalized HMM)

- On 963 human genes:

	<i>Nuc Sens</i>	<i>Nuc Prec</i>	<i>Nuc Acc</i>	<i>Exon Sens</i>	<i>Exon Prec</i>	<i>Exon Acc</i>	<i>Exact Genes</i>
<i>GlimmerHMM</i>	86%	72%	79%	72%	62%	67%	17%
<i>Genscan</i>	86%	68%	77%	69%	60%	65%	13%

- **Note** that overall accuracy is pretty low.

Comparative Methods

Problem: Predict genes in a target genome G based on the contents of G and also based on the contents of one or more informant genomes $I^{(1)} \dots I^{(n)}$:

informant genome: GC-ATCGGTCTTA
 ... | : . | : | . | | : | : | ... } alignment
target genome: ATCGGTAAC-GTGTAAATGC

Rationale: *Natural selection should operate more strongly on protein-coding DNA than on non-functional DNA such as introns.*

Pair Generalized HMMs

Use: find genes simultaneously in 2 genomes
increased signal b/c the structure of homologous genes is often very similar.

- Pair: Each state emits two symbols, one for each sequence
- Generalized Pair: a pair of lengths d, e is drawn from a joint probability distribution and a pair of sequences X, Y of length d, e , respectively, are generated at each state.

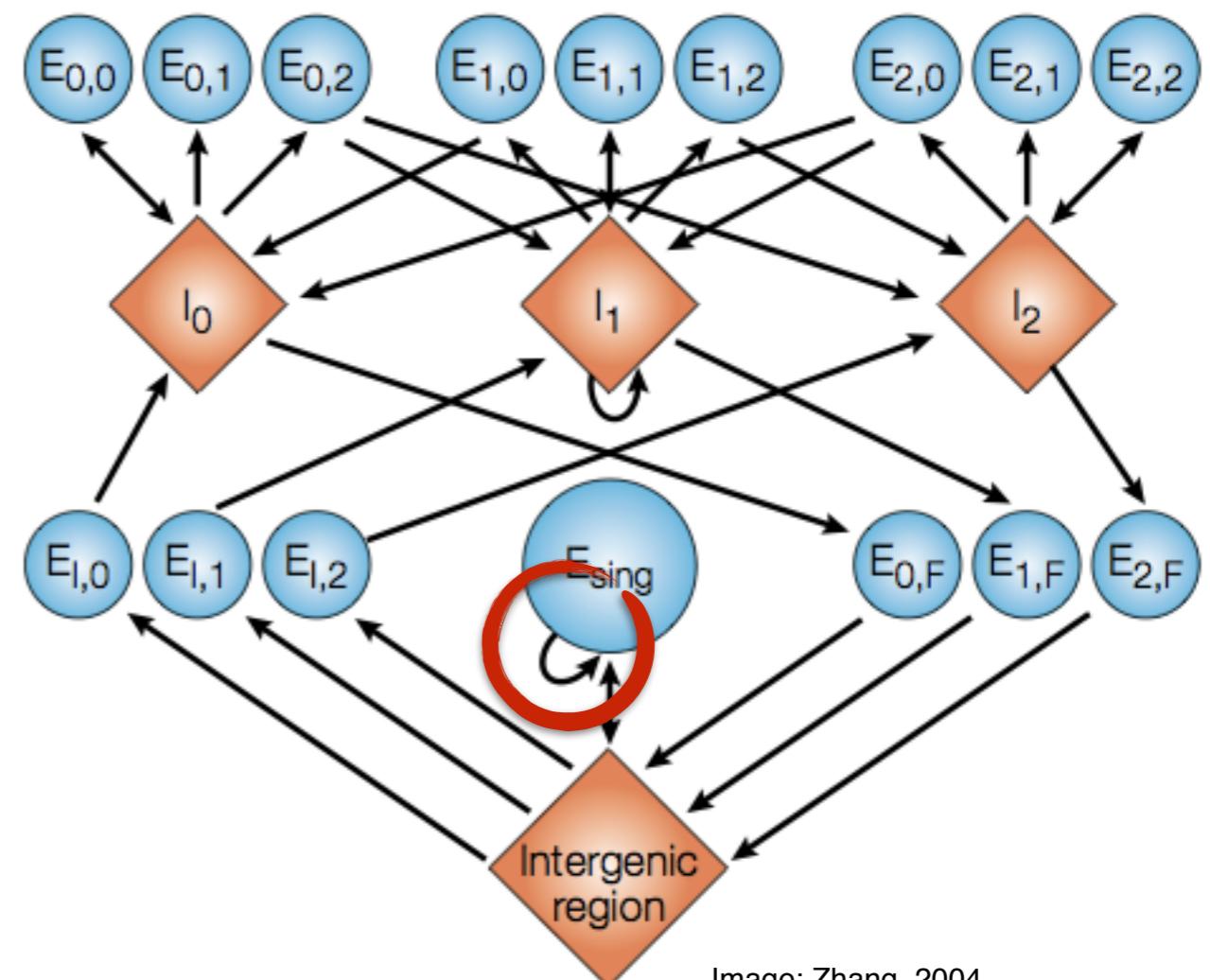


Image: Zhang, 2004

Pachter et al. J Comp Biol, 9(2), 2002

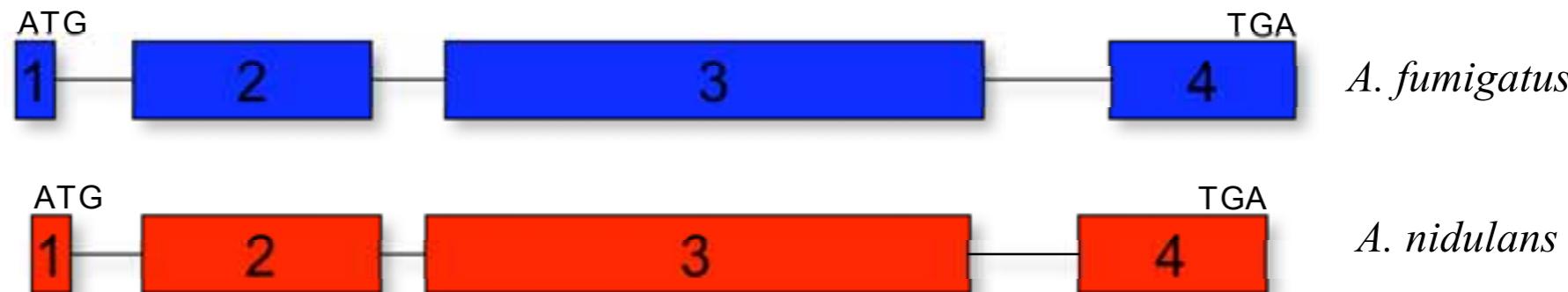
Reverse strand: mirror reflection of above

Accuracy

Data set: 147 high-confidence *Aspergillus fumigatus* × *A. nidulans* orthologs (493 exons, 564kb).

	nucleotide accuracy	exon sensitivity	exon specificity	exact genes
GHMM	99%	78%	73%	54%
GPHMM	99%	89%	85%	74%

How Does Homology Help?



feature	amino acid alignment score	<,>	nucleotide alignment score
exon 1	100%	>	71%
intron 1	14%	<	51%
exon 2	98%	>	85%
intron 2	29%	<	49%
exon 3	97%	>	82%
intron 3	9%	<	49%
exon 4	96%	>	83%

Recap

- Simple gene finding approaches use codon bias and long ORFs to identify genes.
- Many top gene finding programs for Eukaryotes are based on generalizations of Hidden Markov Models because multiple types of signals are present in a gene (intron, exon, etc.)
- Basic HMMs must be generalized in many way (e.g. to emit variable sized strings, to emit two strings simultaneously, etc.).