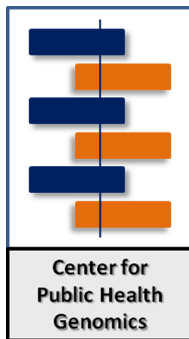


Prokaryotic and Eukaryotic Genome Annotation: gene structure and function

Aaron J. Mackey, Ph.D.

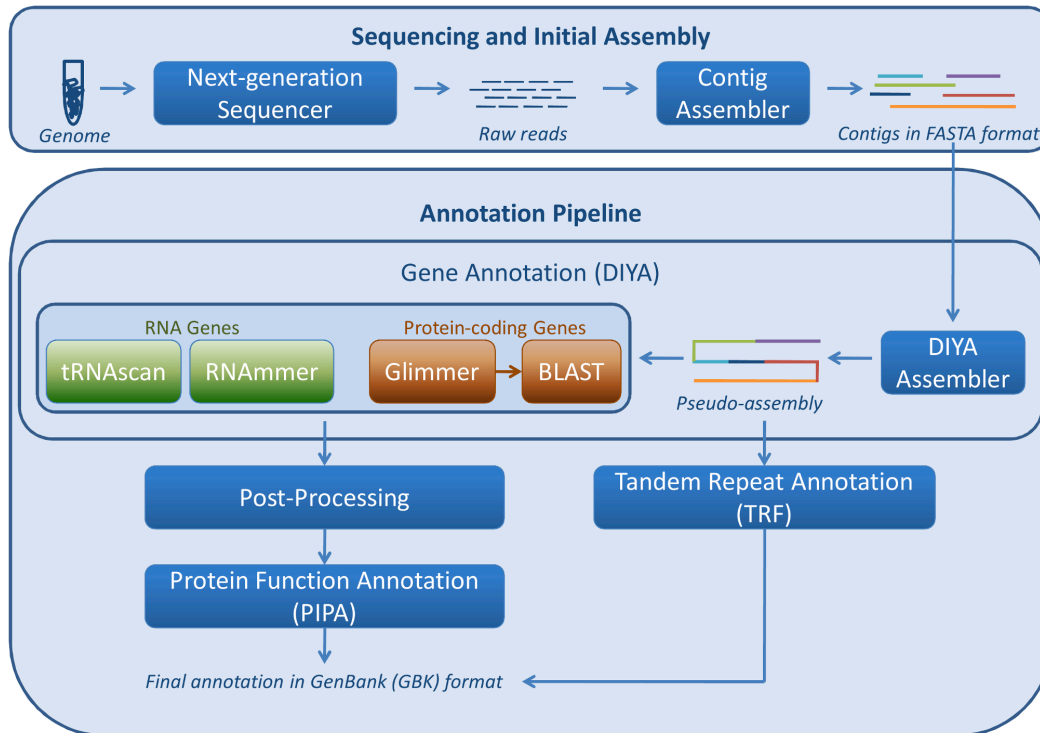
amackey@virginia.edu



Outline

- bacterial gene annotation
- a primer in Hidden Markov Models
- eukaryotic gene annotation

Overview of Bacterial Annotation



how to identify bacterial coding genes:



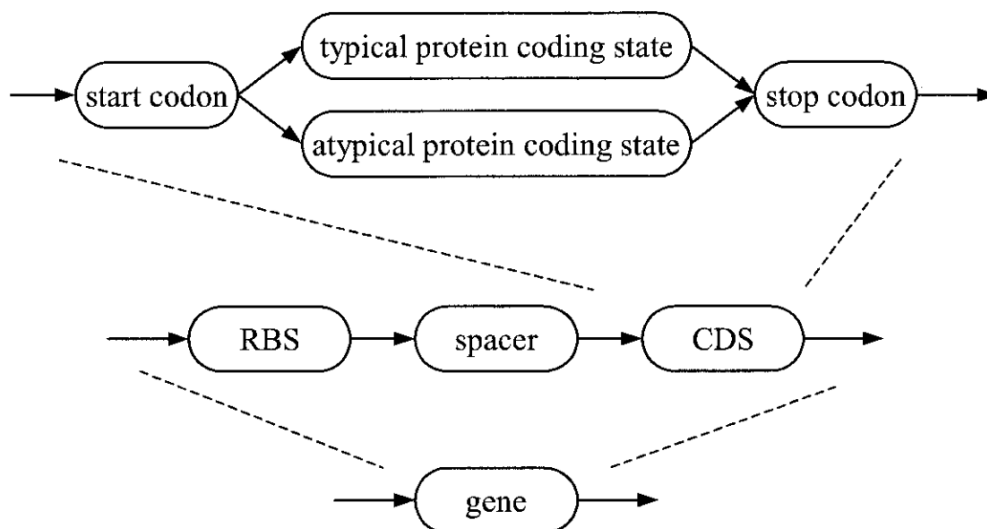
- annotation aims to identify true start (ATG) sites
- “long” open reading frames (ORFs)
 - lots of “short” ORFs missed
- homology to known proteins (BLAST/FASTA)
- Ribosomal binding sites (RBS)
 - canonical Shine-Dalgarno vs. species-specific 16S ribosome
 - SD sequence not required for ribosomal S1 binding at upstream AU sites; requires options
- protein coding potential (codon usage, amino-acid frequency)
 - 3rd-order or 6th-order Hidden Markov Models (HMMs)

ab initio bacterial gene finding



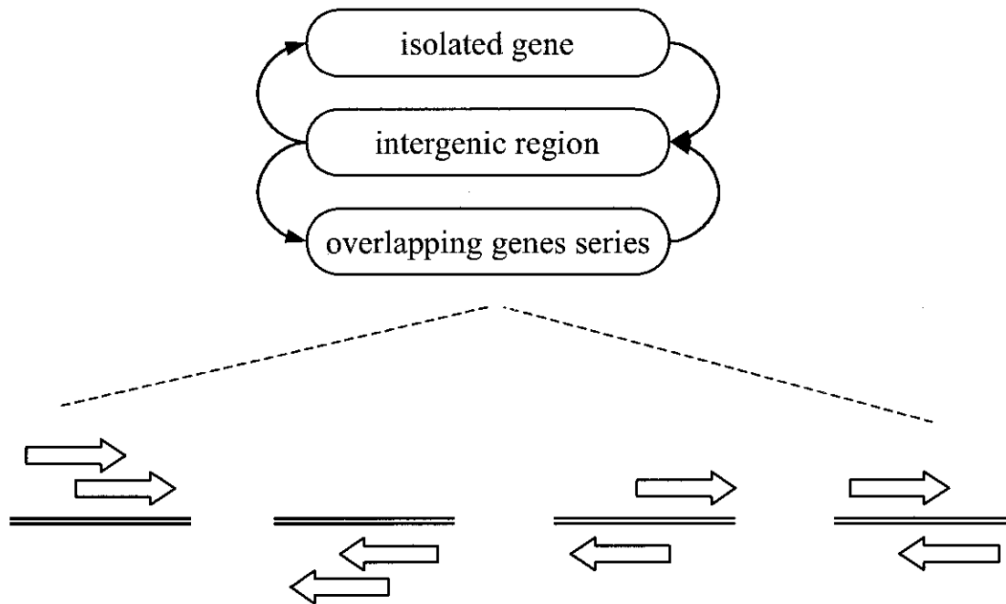
- Glimmer, GeneMark, GeneMark.hmm, GeneMarkS, ORPHEUS, CRITICA
- all *ab initio* methods use some form of statistical models to represent expected microbial gene structures.

GeneMark.hmm's microbial gene grammar

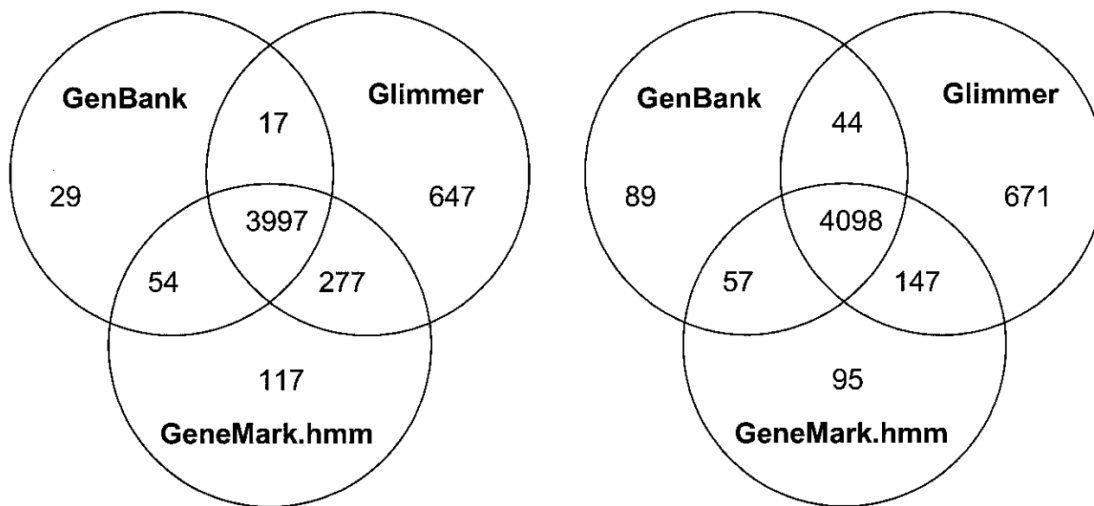


more about Hidden Markov Models (HMMs) soon!

GeneMark.hmm's grammar for overlapping/operon genes



bacterial gene finders are mostly accurate



A *B. subtilis*

B *E. coli*

Glimmer generally more sensitive (false positives?)

bacterial gene finders are mostly accurate

Table 4. Comparison of the GeneMarkS, Glimmer 2.02 and ORPHEUS gene prediction programs on the following test sets: the *B.subtilis* genome as annotated in GenBank (A); three sets of *B.subtilis* genes shorter than 300 nt with at least one (B), at least two (C) and at least 10 (D) significant homologies determined by BLAST analysis; and a set of 195 experimentally validated *E.coli* genes (E)

Program	Test set	Genes in test set	Genes precisely predicted ^a	Genes detected ^b (3' end)
Glimmer	A	4099	2556 (62.4%)	4023 (98.1%)
ORPHEUS	A		3028 (73.9%)	3484 (85.0%)
GeneMarkS	A		3412 (83.2%)	3962 (96.7%)
Glimmer	B	123	70 (57.0%)	112 (91.1%)
GeneMarkS	B		102 (82.9%)	113 (91.9%)
Glimmer	C	72	41 (57.0%)	66 (91.7%)
GeneMarkS	C		64 (88.9%)	68 (94.4%)
Glimmer	D	51	26 (51.0%)	45 (88.2%)
GeneMarkS	D		46 (90.2%)	48 (94.1%)
Glimmer	E	195	139 (71.3%)	195 (100%)
ORPHEUS	E		148 (75.9%)	181 (92.8%)
GeneMarkS	E		184 (94.4%)	195 (100%)

Numbers in bold indicate the highest number of genes detected or genes precisely predicted for each test set.

^aRefers to the case where both the 5' end and the 3' end predictions match the annotation.

^bRefers to the case where the 3' end prediction (and not necessarily 5' end prediction) matches the annotation.

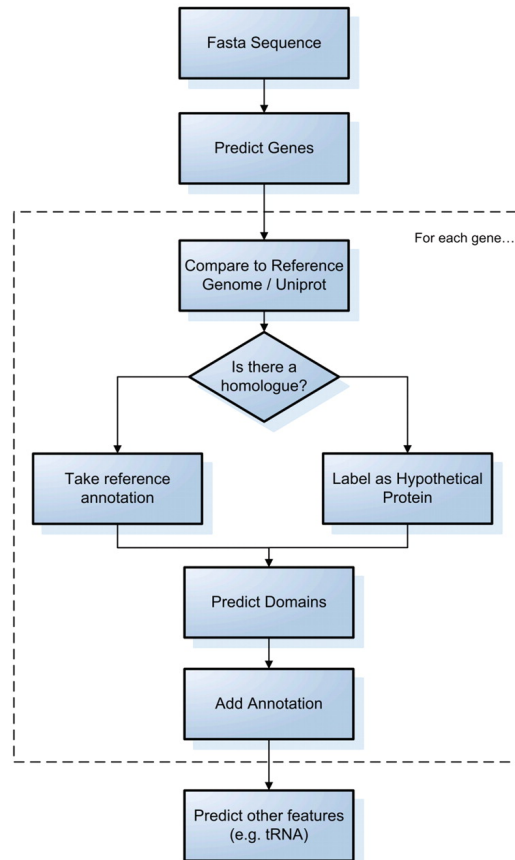
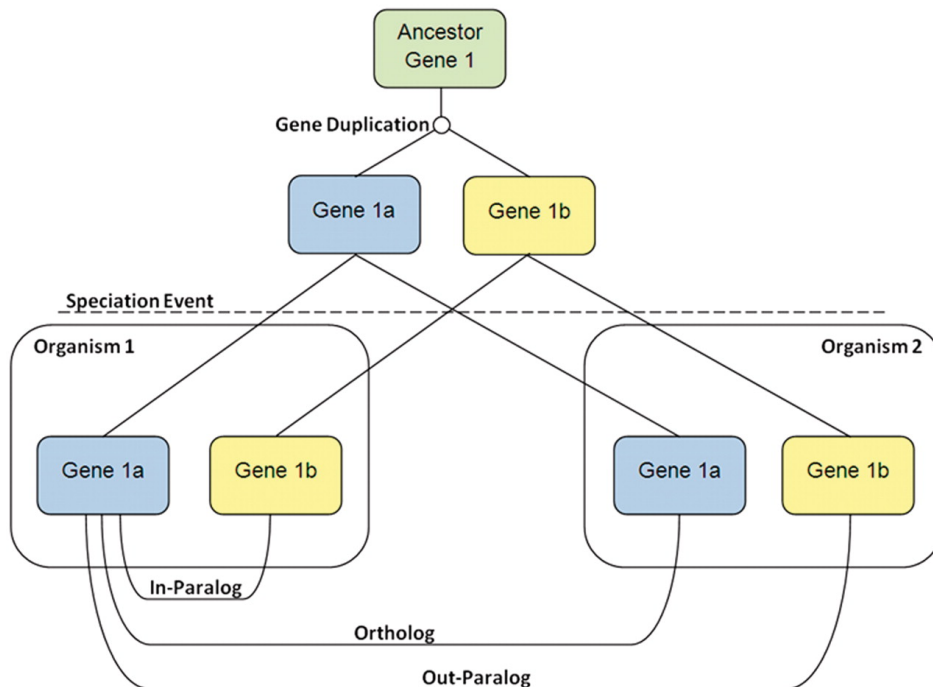


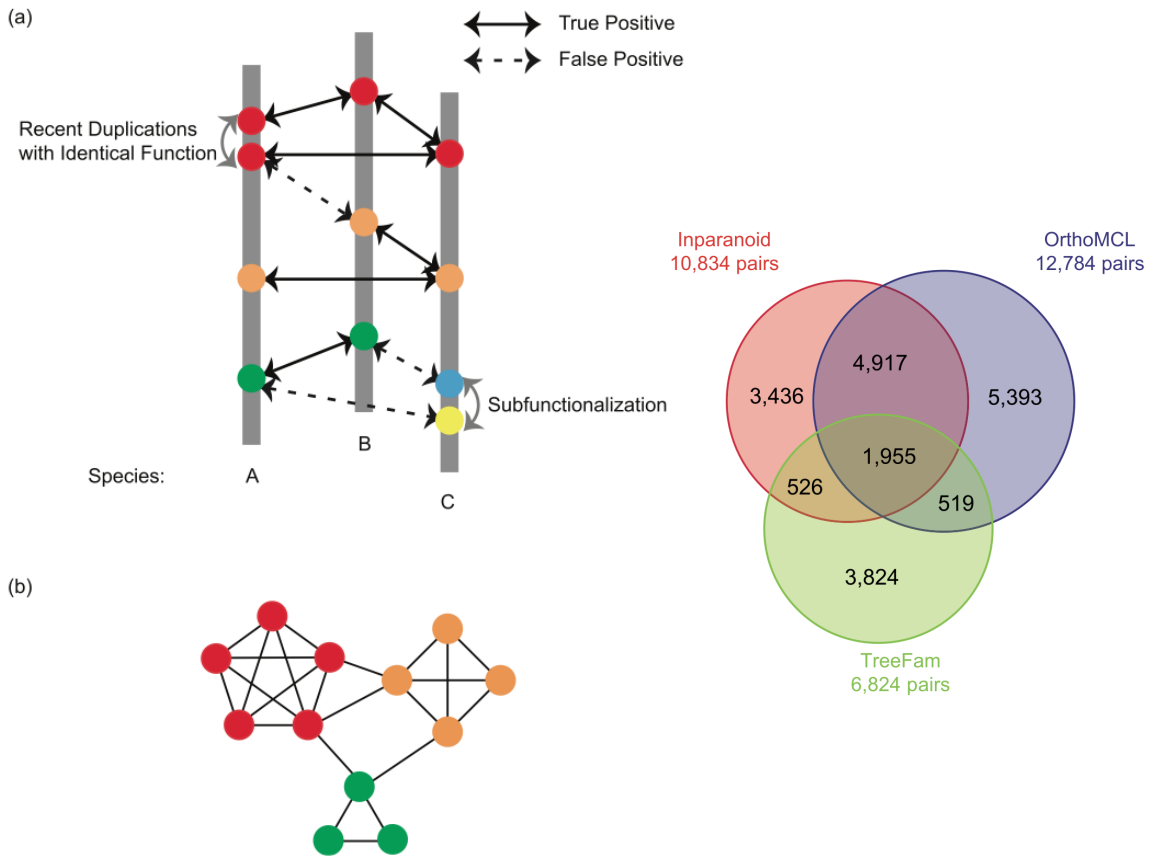
Table 3. Comparison of annotation of *E.coli K-12* accession U00096.2

Feature	Reference	Prokka	RAST	xBase2
Total CDS	4321	4305	4512	4444
<i>Matching start</i>	–	3828	3571	3025
<i>Different start</i>	–	318	533	1052
<i>Missing CDS</i>	–	172	214	241
<i>Extra CDS</i>	–	159	405	367
<i>Hypothetical protein</i>	18	276	638	156
<i>With EC number</i>	1114	1050	1118	0
Total tRNA	89	88	86	88
Total rRNA	22	22	22	22

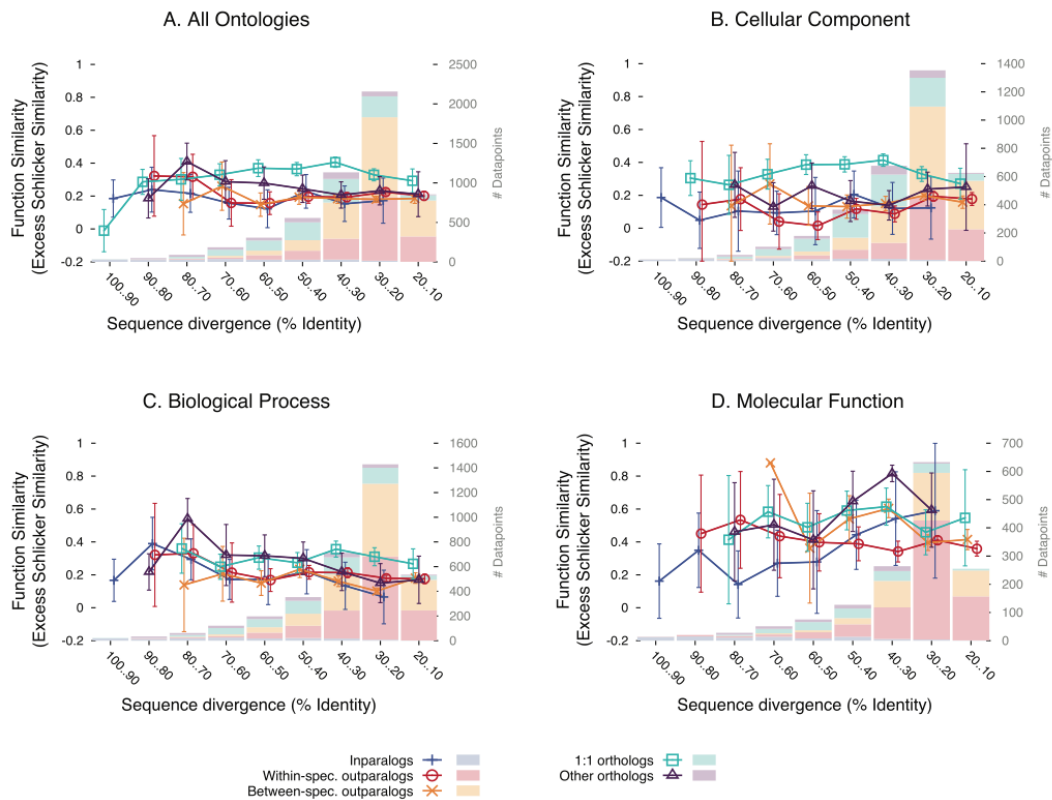
The bold denotes the best performing tool (column) for that attribute (row). The italics are “subsets” of the “Total CDS” section.

orthology-dependent annotation





Functional Similarity of Orthologs and Paralogs in *S. cerevisiae* and *S. Pombe*



take home: bacterial gene annotation

- 5' ATG start sites harder to get right than 3' stop sites
- homology-based methods are complementary to *ab initio* tools
- functional prediction driven by homology and existing annotations: "guilt by association"
- integrated annotation pipelines (AGeS, RAST, PIPA, MaGe, Prokka, JCVI/IGS annotation service) are the modern standard
- incomplete/metagenomic assemblies still rife with sequencing+assembly errors ... impact on ORFs
- OK, but what is this HMM stuff all about?

"What makes HMMs so popular is that the name is so tantalizing. Something is hidden, and we're finding it, and we have a Russian name to do it."

- David Lipman
Science: 273:590, 1996

Hidden Markov Models (HMMs)

- a statistical *model* that relates *observations* to underlying, explanatory *variables*
 - a linear model relates y to x_1, x_2, \dots, x_n with:
$$y = a + b_1x_1 + b_2x_2 + \dots + b_nx_n + \varepsilon$$
- the observations D are sequential (**Markov**), exhibit K^{th} -order (e.g. 1st-order) correlations
 - usually shown as edges between nodes in a graph
- all x_i 's (for some subset of i in n) are structurally unobserved, latent, i.e. **hidden**
 - not the same as “missing data”
 - only categorical variables, i.e. “labels”

HMMs for sequential inference

- four aspects/parts to all HMMs:
 - observed sequential data (D)
 - hidden/unobserved labels (L)
 - state-graph topology/structure (G)
 - enumerated states (nodes)
 - allowed transmissions (edges) between states
 - labeled state emissions (observations)
 - model parameters (θ_G)
 - transmission & emission probabilities

HMMs for sequential inference

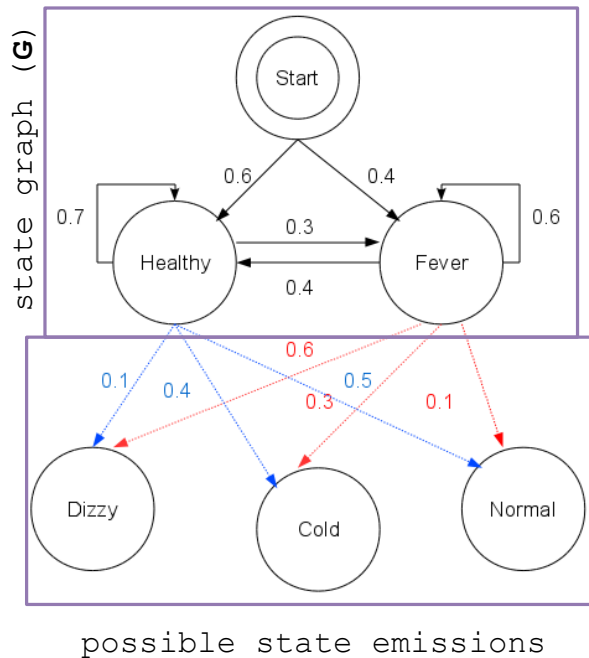
- four aspects/parts to all HMMs:
 - observed sequential data (D)
 - hidden/unobserved labels (L)
 - state-graph topology/structure (G)
 - model parameters (θ_G)
- four issues answered with HMMs:
 - given G, D, L, θ_G ; how likely is D (scoring)?
 - given G, D, θ_G ; what is the best L? (labeling)
 - given G, D, L ; what is the best θ_G ? (training)
 - given G, D ; what is the best θ_G ? (training)

HMM example: the sick child

- a child feels either “cold”, “dizzy” or “normal” at any given time (observed)
- the parent is trying to figure out whether the child is “healthy” or “feverish” (hidden labels)
- being “cold”, “dizzy”, or “normal” does not *directly* indicate health/fever, ***but is correlated***
- health/fever episodes are *sequentially correlated*

completely specified sick child HMM:

(hidden) labels (**L**): H H H **F F** H H H H **F F F** H H H H
 data (**D**): C N N C D N N D C N C D N C N N



four aspects/parts to all HMMs:

- observed sequential data (**D**)
- hidden/unobserved labels (**L**)
- state-graph topology/structure (**G**)
- model parameters (θ_G)
 - emission alphabet (**D**izzy, **C**old, **N**ormal)
 - state-specific emission probabilities (red and blue numbers)
 - state-to-state transition probabilities (black numbers)

HMMs for sequential inference

- given G, D, L, θ_G ; how likely is D ? (scoring)
 - calculate $P(D|L)$ using Markov chain rule
- given G, D, θ_G ; what is the best L ? (labeling)
 - employ Viterbi along state/observation “trellis”
- given G, D, L ; what is the best θ_G ?
 (training with labels/truth: supervised)
 - maximum likelihood (ML): find θ_G that optimizes $P(D|L)$
- given G, D ; what is the best θ_G ?
 (training without labels/truth: unsupervised)
 - Baum-Welch (EM): iterate between expected labeling (forward/backward) and training (ML) until convergence

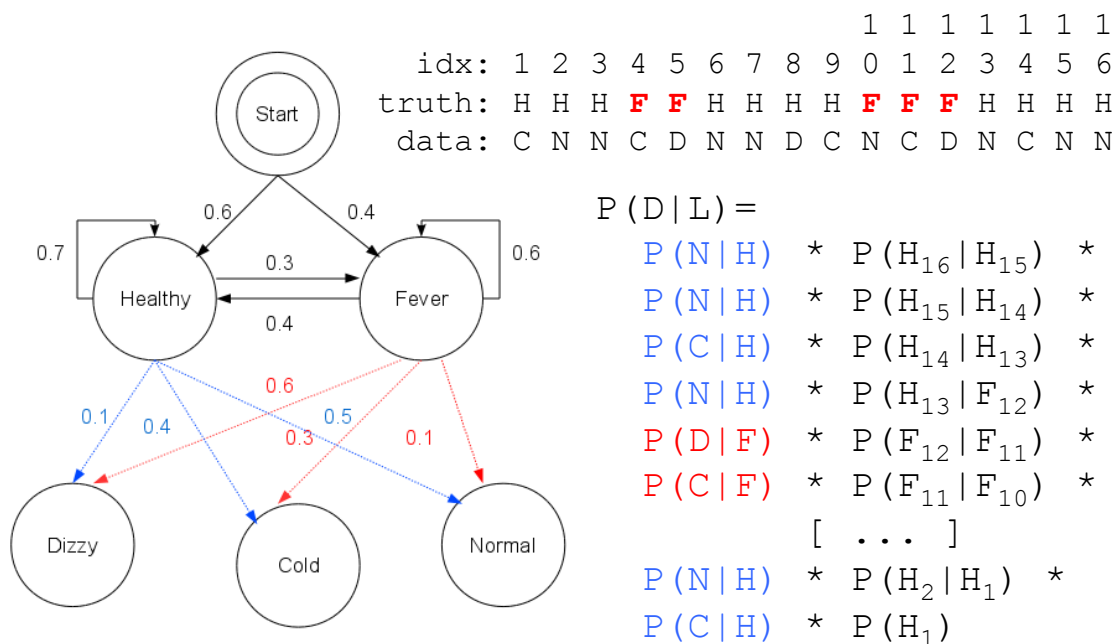
Basic conditional probability rule:

$$P(A, B) = P(A|B)P(B)$$

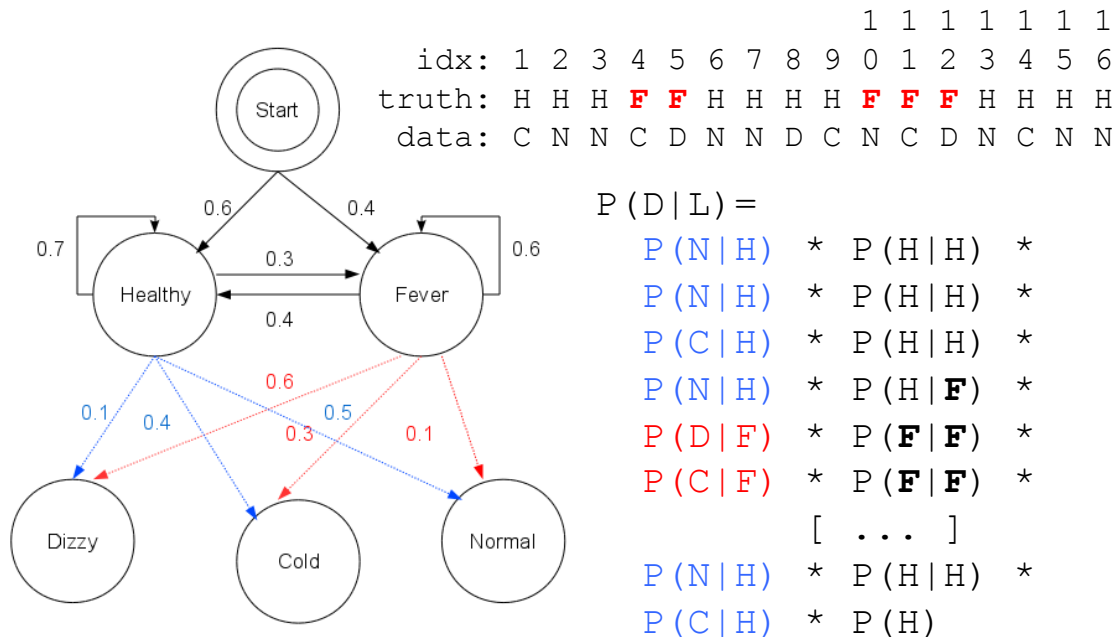
The Markov chain rule:

$$\begin{aligned} P(q_1, q_2, \dots, q_T) &= P(q_T | q_1, q_2, \dots, q_{T-1}) P(q_1, q_2, \dots, q_{T-1}) \\ &= P(q_T | q_{T-1}) P(q_1, q_2, \dots, q_{T-1}) \\ &= P(q_T | q_{T-1}) P(q_{T-1} | q_{T-2}) P(q_1, q_2, \dots, q_{T-2}) \\ &= P(q_T | q_{T-1}) P(q_{T-1} | q_{T-2}) \cdots P(q_2 | q_1) P(q_1) \end{aligned}$$

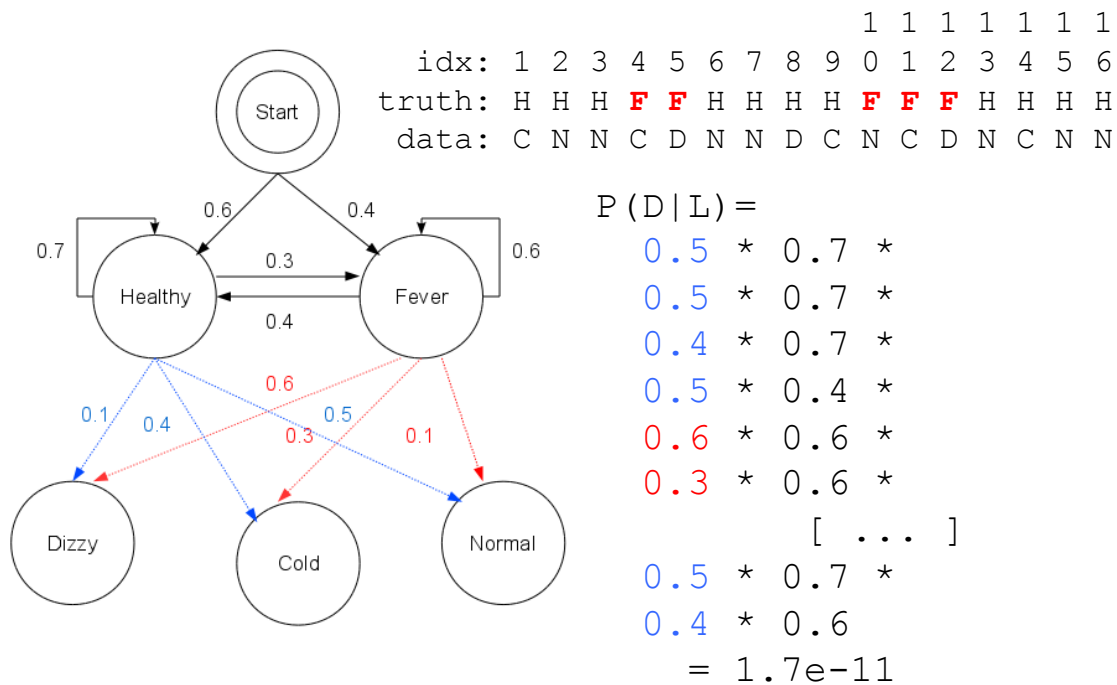
HMM scoring: Markov chain rule



HMM scoring: Markov chain rule



HMM scoring: Markov chain rule



High Scoring != High Probability

truth: H H H **F F** H H H H **F F F** H H H H
data: C N N C D N N D C N C D N C N N

1.7e-11 is not very probable; how
“remarkable” is this particular set of
observations, compared to a more “expected”
series of observations?

truth: H H H **F F** H H H H **F F F** H H H H
data: N N N D D N N N N D D D N N N N

(this is the “perfect” series of observations with
maximal correlation to the underlying truth)

High Scoring != High Probability

truth: H H H **F F** H H H H **F F F** H H H H
data: C N N C D N N D C N C D N C N N

1.7e-11 is not very probable; how
“remarkable” is this particular set of
observations, compared to a more “expected”
series of observations?

truth: H H H **F F** H H H H **F F F** H H H H
data: N N N D D N N N N D D D N N N N

Answer: 4.1e-09 – more than 200x more likely
that the observed data, but not itself high
probability -> large combinatoric space of
possible sequences

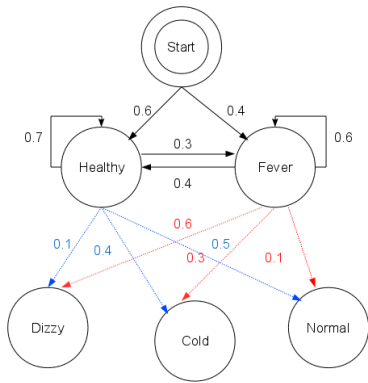
overall probability of a sequence

- instead of $P(D|L, G, \theta_G)$, we could ask $P(D|G, \theta_G)$ – i.e. independent of any “true” labeling, what’s the chance of this exact sequence to arise from this HMM?
- the “forward” algorithm calculates this probability:
 - original sequence: $P(D|G, \theta_G) = 1.9e-08$
 - “expected” sequence: $2.5e-08$

HMMs for sequential inference

- given G, D, L, θ_G ; how likely is D ? (scoring)
 - calculate $P(D|L)$ using Markov chain rule
- given G, D, θ_G ; what is the best L ? (labeling)
 - employ Viterbi along state/observation “trellis”
- given G, D, L ; what is the best θ_G ?
(training with labels/truth: supervised)
 - maximum likelihood (ML): find θ_G that optimizes $P(D|L)$
- given G, D ; what is the best θ_G ?
(training without labels/truth: unsupervised)
 - Baum-Welch (EM): iterate between expected labeling (forward/backward) and training (ML) until convergence

HMM labeling: Viterbi



```

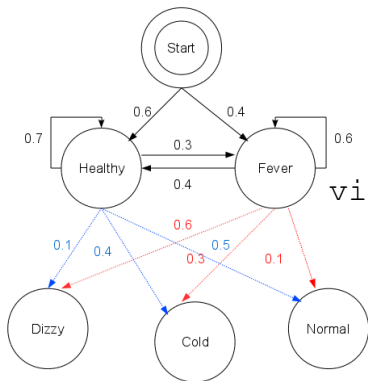
idx: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
truth: H H H F F H H H H F F F H H H H
data: C N N C D N N D C N C D N C N N
    
```

$$P_{1,j} = P(L_1 | S_{1,j})P(S_{1,j})$$

$$P_{i,j} = \max_{k \in K} \{ P(L_i | S_{i,j})P(S_{i,j} | S_{i-1,k})P_{i-1,k} \}$$

Data:	C	N	N	C	D	N	N	D	C	N	C	D	N	...
Healthy	0.24	0.084 (H)	$P(N H) * P(H H) * 0.084$ $P(N H) * P(H F) * 0.0072$											
Fever	0.12	0.0072 (F H)	$P(N F) * P(F H) * 0.084$ $P(N F) * P(F F) * 0.0072$											
Label:	H	H	H	H	F	H	H	F	H	H	?	?	?	?

HMM labeling: Viterbi



```

idx: 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
truth: H H H F F H H H H F F F H H H H
data: C N N C D N N D C N C D N C N N
viterbi: H H H H F H H F H H H F H H H H
    
```

$$P_{1,j} = P(L_1 | S_{1,j})P(S_{1,j})$$

$$P_{i,j} = \max_{k \in K} \{ P(L_i | S_{i,j})P(S_{i,j} | S_{i-1,k})P_{i-1,k} \}$$

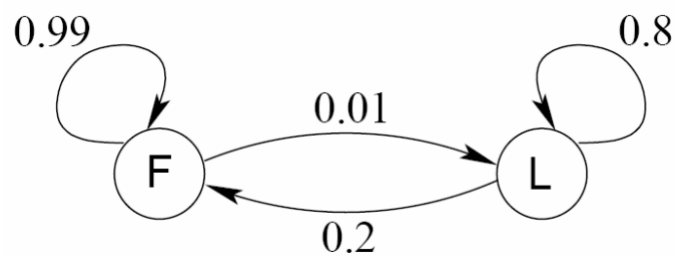
Data:	C	N	N	C	D	N	N	D	C	N	C	D	N	...
Healthy	0.24	0.084 (H)	0.029 (H)	0.0082 (H)	5.8e-04 (H)	3e-04 (F)	1e-04 (H)	7.3e-06 (H)	3e-06 (F)	1e-06 (H)	2.9e-07 (H)	2e-08 (H)	1.1e-08 (F)	
Fever	0.12	0.0072 (F H)	0.0025 (H)	0.0026 (H)	0.0015 (H)	8.9e-05 (F)	8.9e-06 (H)	1.9e-05 (H)	3.4e-06 (F)	2e-07 (H)	9.4e-08 (H)	5.3e-08 (H)	3.2e-09 (F)	
Label:	H	H	H	H	F	H	H	F	H	H	H	F	H	...

traceback highest scoring path ...

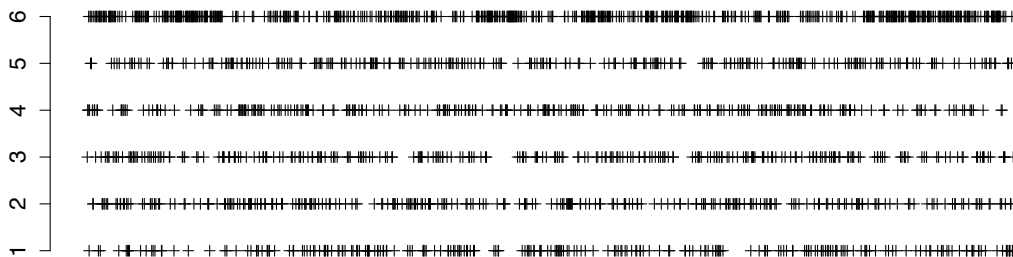
The occasionally dishonest casino

A casino uses a fair die most of the time, but occasionally switches to a loaded one:

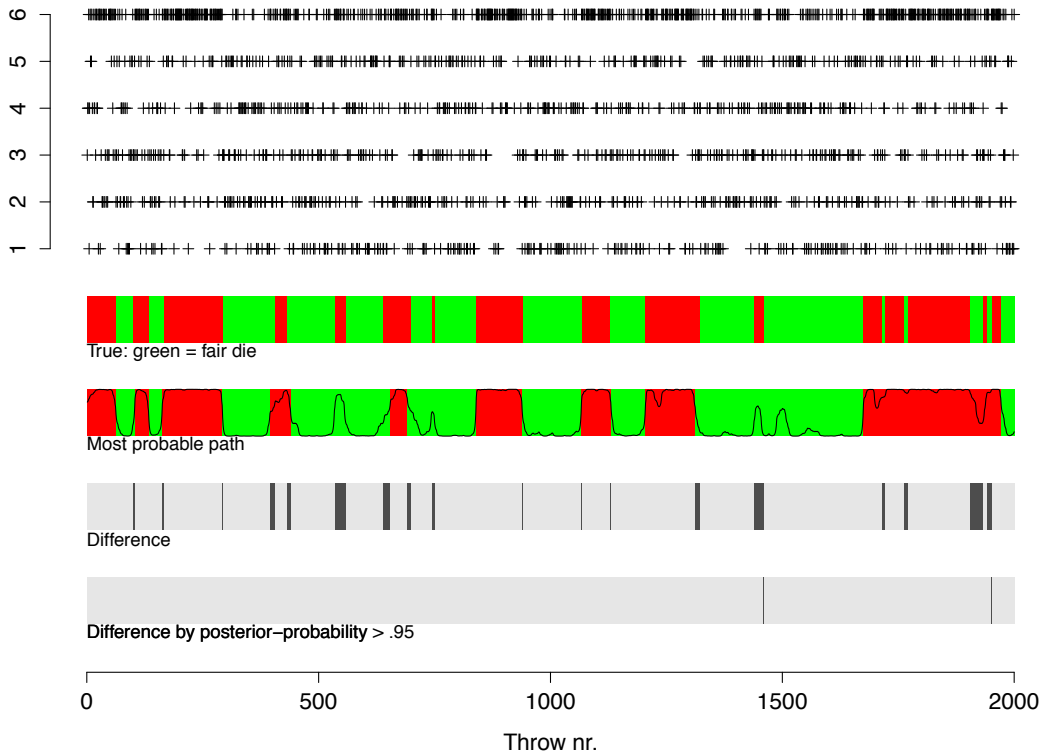
- Fair die: $\text{Prob}(1) = \text{Prob}(2) = \dots = \text{Prob}(6) = 1/6$
- Loaded: $\text{Prob}(1) = \text{Prob}(2) = \dots = \text{Prob}(5) = 1/10$, but $\text{Prob}(6) = 1/2$



Fair and unfair die



Fair and unfair die

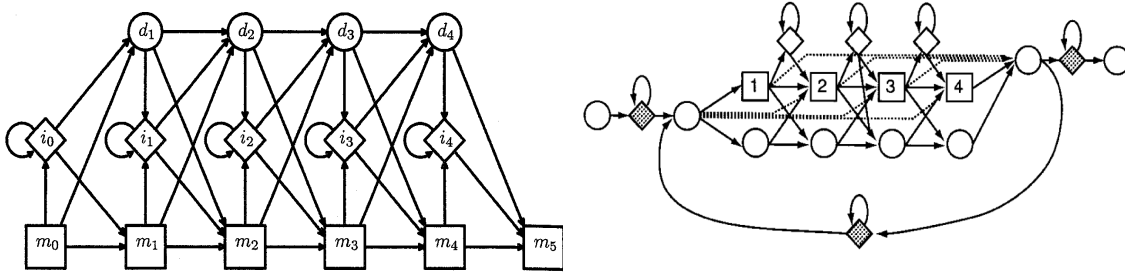


HMMs for sequential inference

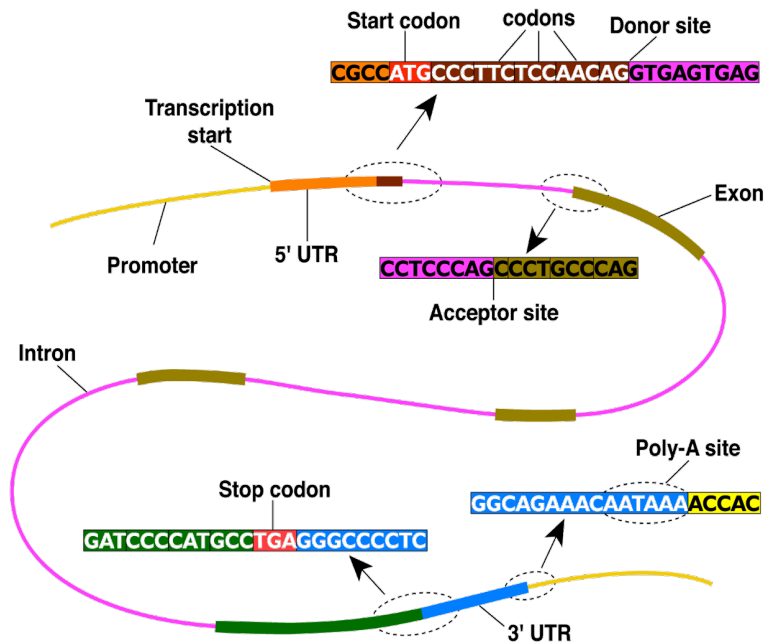
- given G, D, L, θ_G ; how likely is D ? (scoring)
 - calculate $P(D|L)$ using Markov chain rule
- given G, D, θ_G ; what is the best L ? (labeling)
 - employ Viterbi along state/observation “trellis”
- given G, D, L ; what is the best θ_G ?
(training with labels/truth: supervised)
 - maximum likelihood (ML): find θ_G that optimizes $P(D|L)$
- given G, D ; what is the best θ_G ?
(training without labels/truth: unsupervised)
 - Baum-Welch (EM): iterate between expected labeling (forward/backward) and training (ML) until convergence

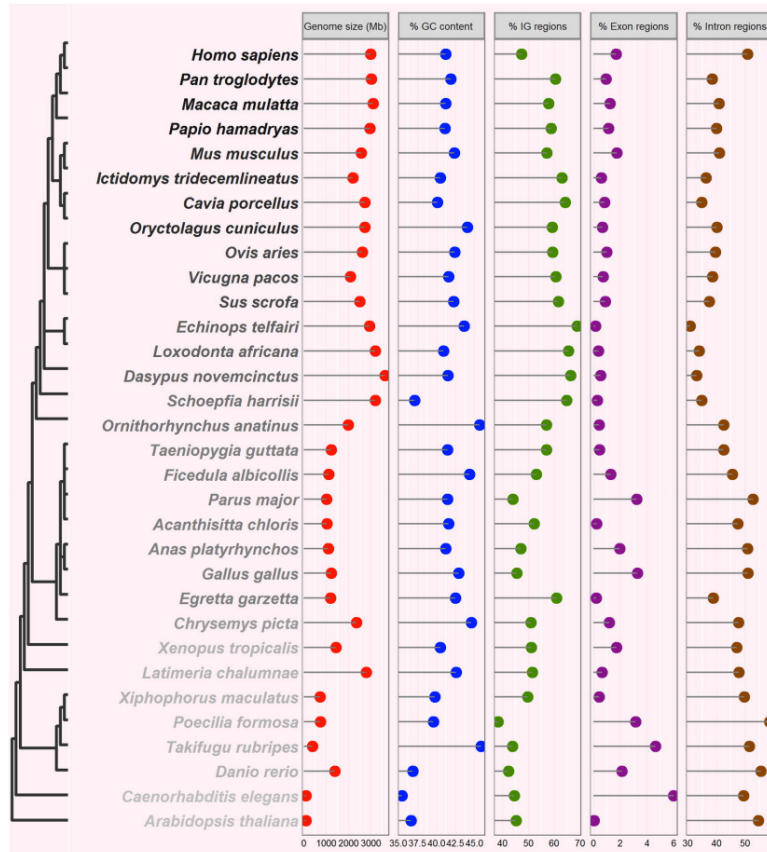
HMMs in computational genomics

- protein domain/sequence alignment
- multiple sequence alignment
- CNV inference
- SNP/haplotype inference
- CpG-methylation inference
- many, many, many others

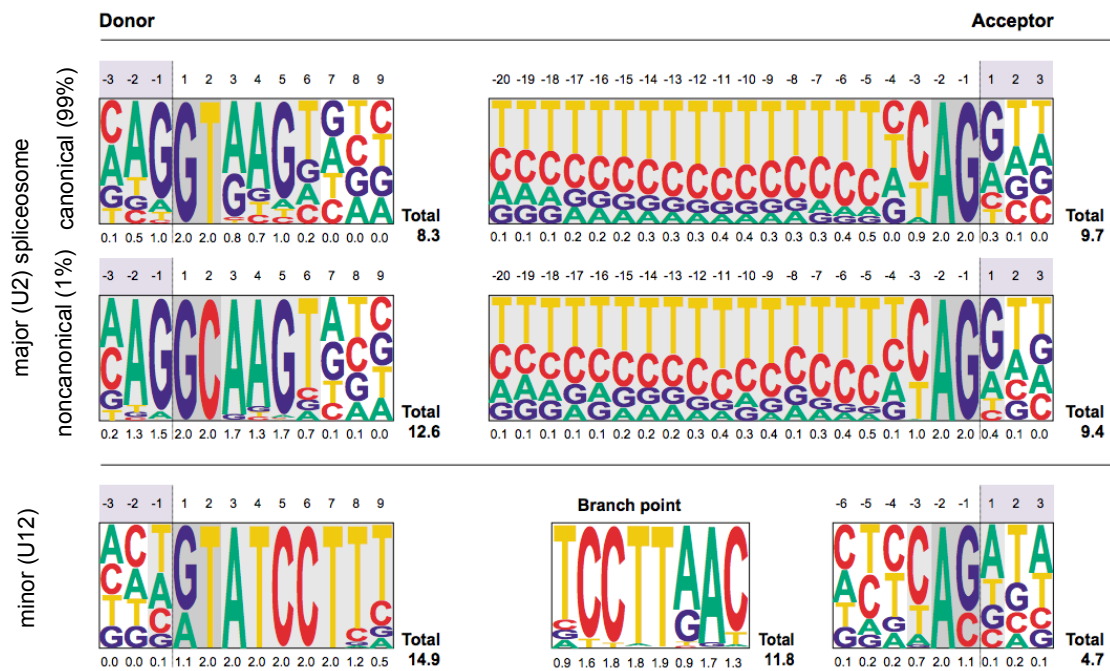


eukaryotic gene structure

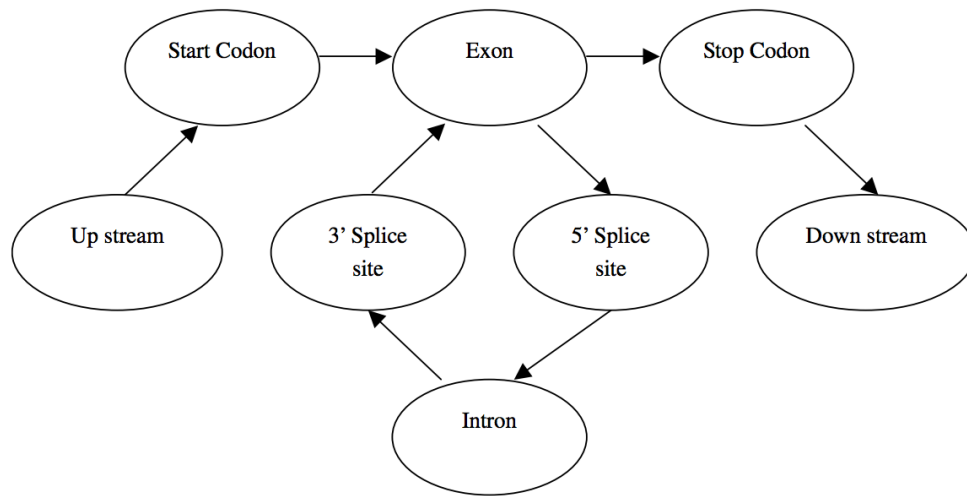




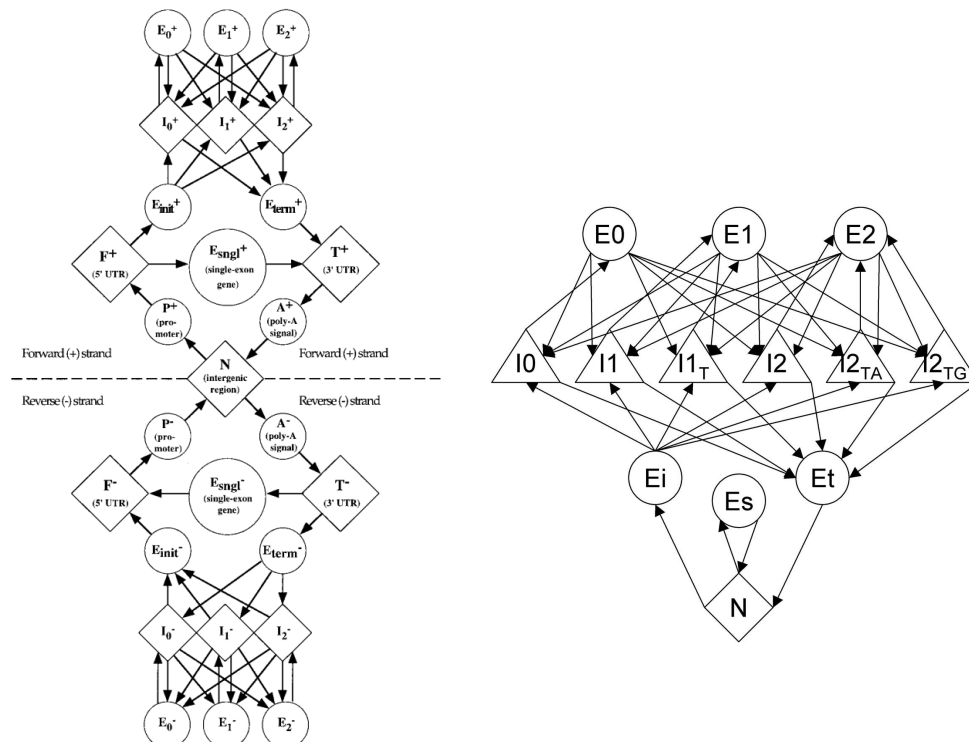
Intron Sequence Patterns



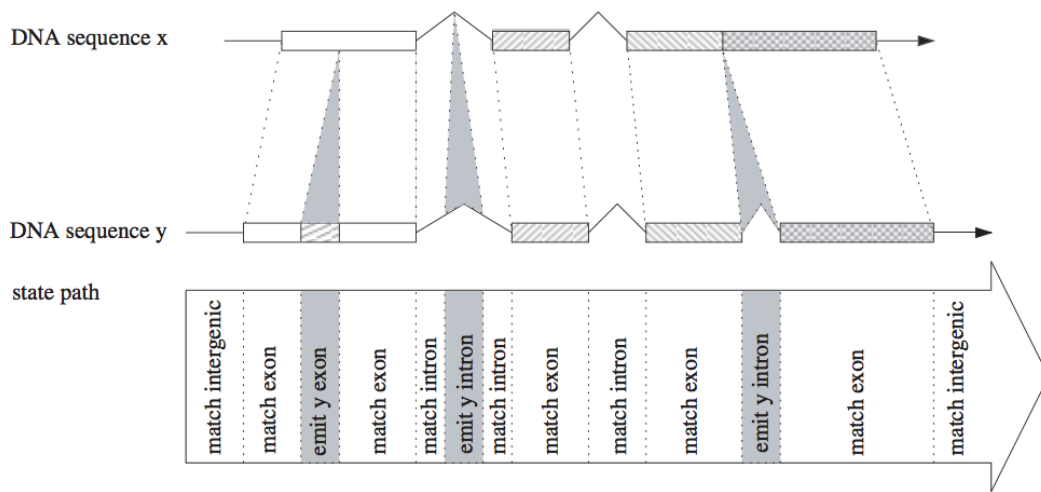
a simple HMM for eukaryotic genes



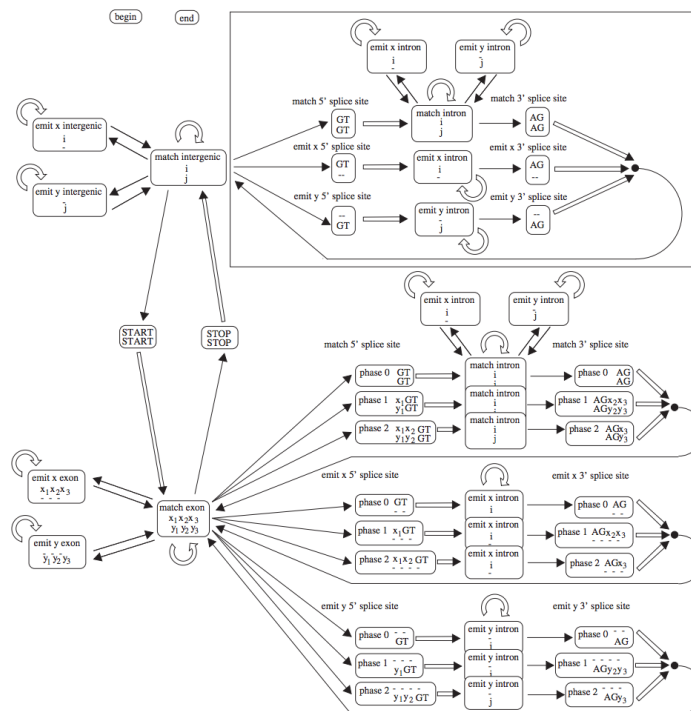
the GenScan/SNAP HMM topology



Comparative *ab initio* methods



DoubleScan



TwinScan/N-Scan

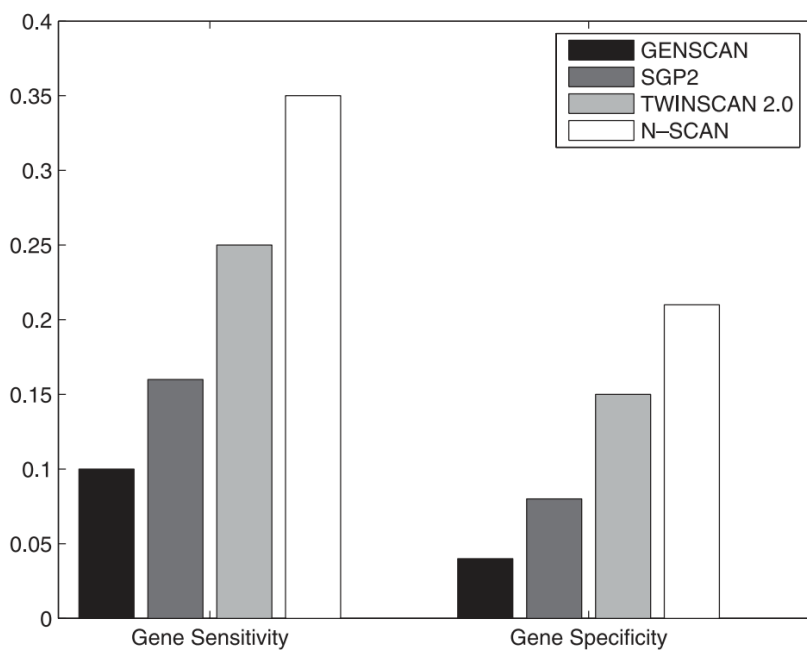
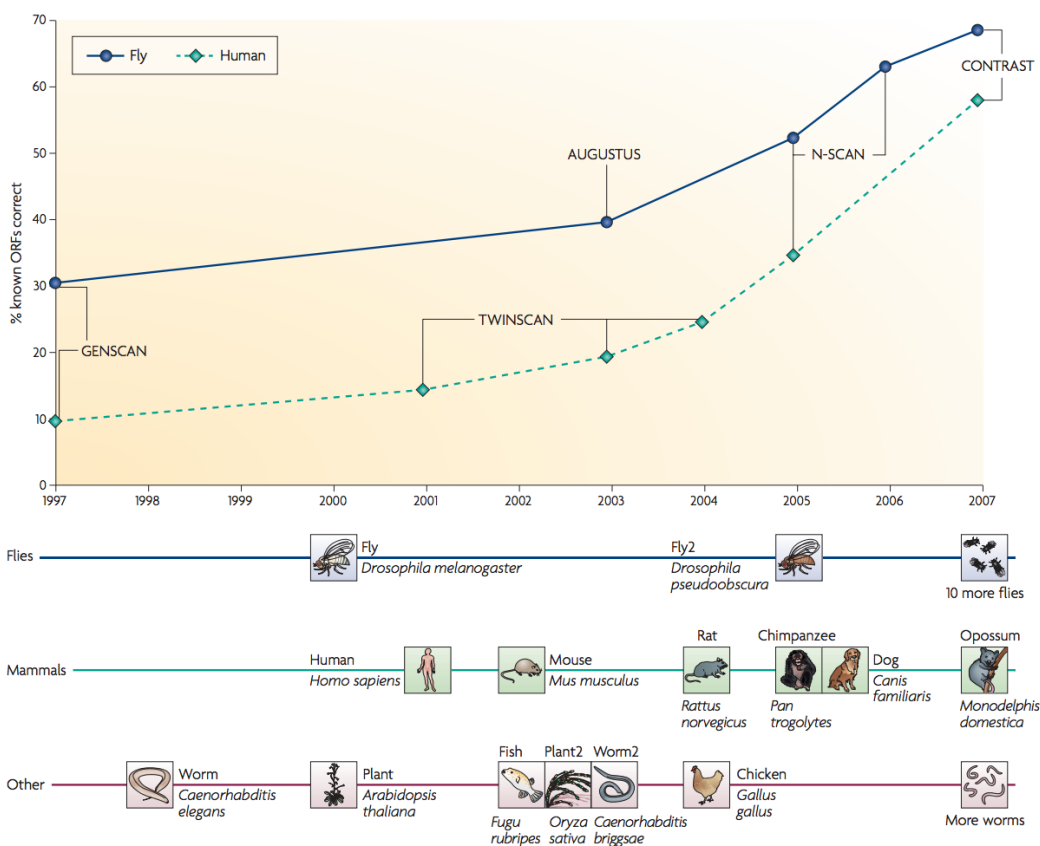


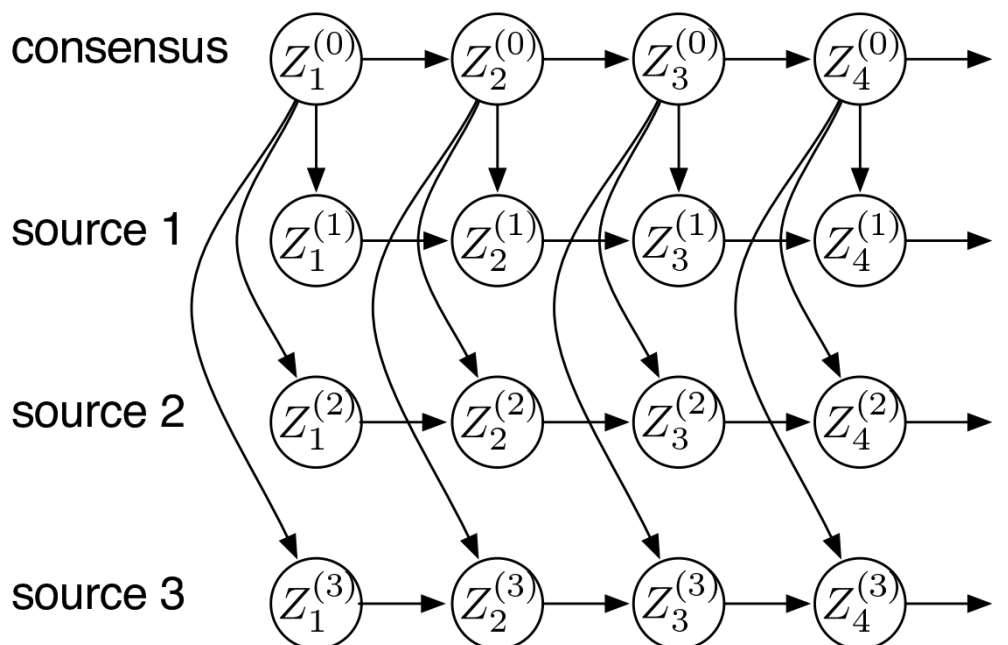
FIG. 4. Exact gene accuracy in human.



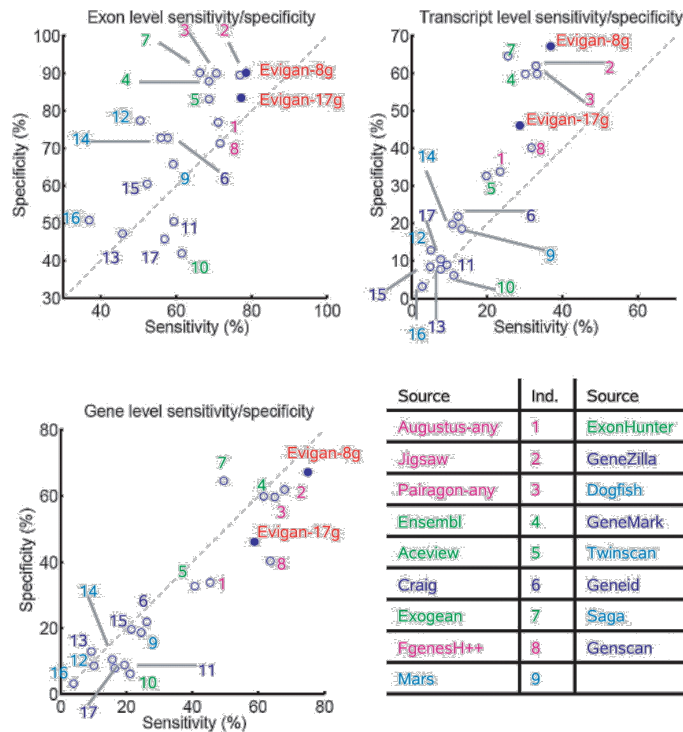
“combiners”

segment index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	alphabet
consensus (0)	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	{E0, E1, E2, X, I0, I1-T, I1-X, I2-TA, I2-TX, I2-XX}
gene finder (1)	X	E1	E1	E1	E1	E1	I	I	E1	E1	E1	E1	E1	E1	E1	X	{E0, E1, E2, X, I}
gene finder (2)	X	X	X	E1	E1	E1	E1	I	I	I	E1	E1	E1	E1	E1	X	{E0, E1, E2, X, I}
gene finder (3)	X	E1	E1	E1	E1	E1	E1	I	I	E1	E1	E1	E1	E1	E1	X	{E0, E1, E2, X, I}
splice predictor (4)	U	U	U	I	E	U	E	I	I	E	I	U	E	E	I	U	{E, I, U}
transcript alignment (5)	XI	XI	E	E	E	XI	XI	XI	XI	XI	E	E	E	E	E	XI	{E, XI}
protein alignment (6)	XI	XI	XI	XI	XI	XI	XI	XI	XI	E0	E0	E0	E0	E0	E0	XI	{E0, E1, E2, XI}
EST (7)	X	X	E	E	E	E	E	I	I	E	E	E	E	X	X	X	{E, X, I}

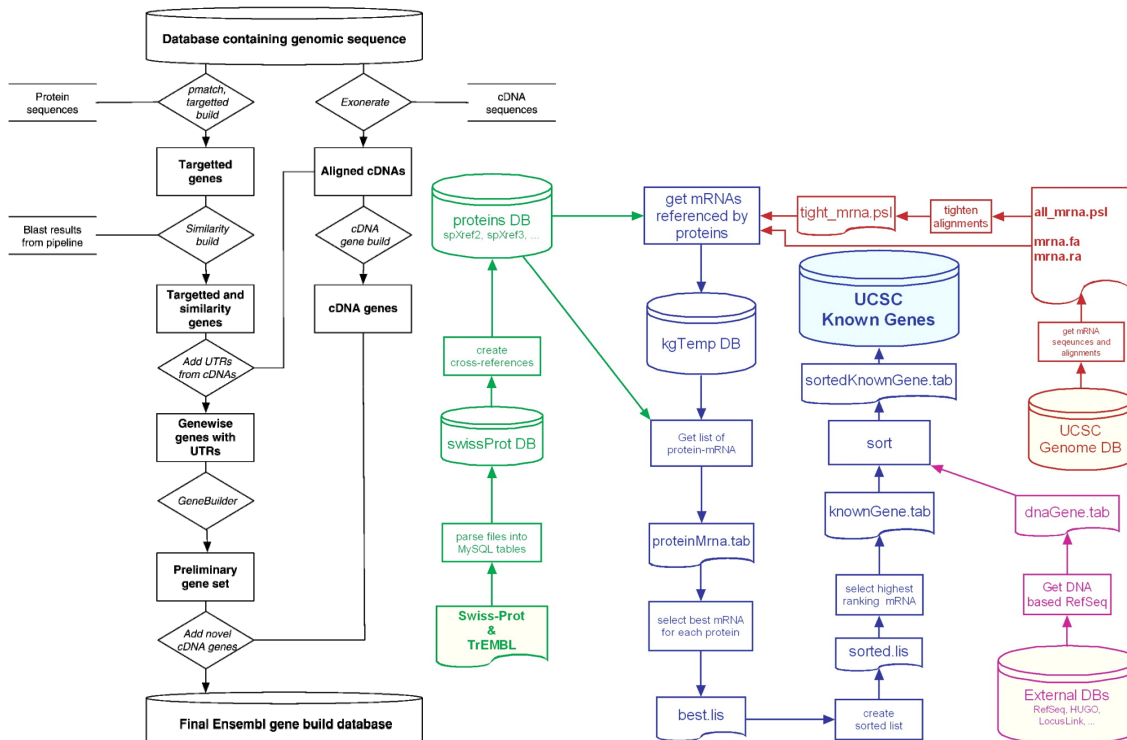
“combiners” – just another HMM



combiners improve accuracy a bit



Ensembl and UCSC pipelines



Alternative Splicing

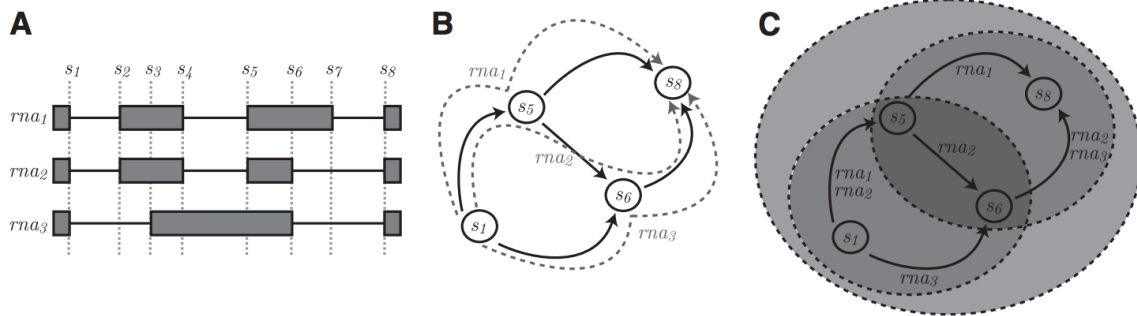
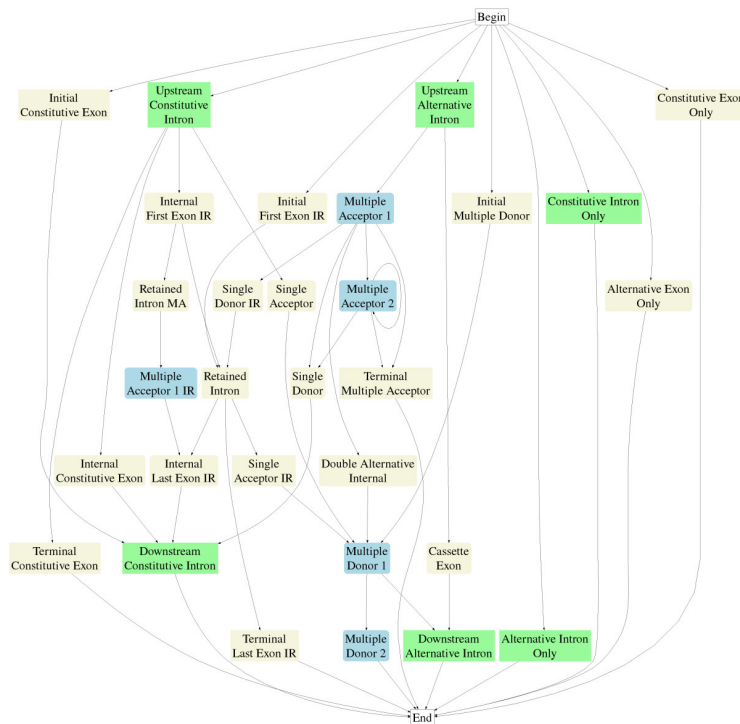


FIG. 2. (A) A cutoff from a locus showing $k=3$ transcripts (rna_1 , rna_2 and rna_3) and 8 sites (s_1, \dots, s_8). The exon-intron structure is shown schematically, i.e., exons (boxes) and introns (lines) are not drawn to scale. Different variants can be observed, for instance ($s_1, s_5, \{rna_1, rna_2\}$). (B) The corresponding splicing graph structure after contracting uninformative vertices. Dotted lines indicate the paths supported by single transcripts rna_1 , rna_2 and rna_3 . (C) Ovals highlight all 3 bubbles, that is ($s_1, s_6, \{rna_2\}$), ($s_5, s_8, \{rna_1, rna_2\}$) and ($s_1, s_8, \{rna_1, rna_2, rna_3\}$). In contrast, there exists no bubble between s_5 and s_6 because they are connected by only a single variant (i.e., rna_2).

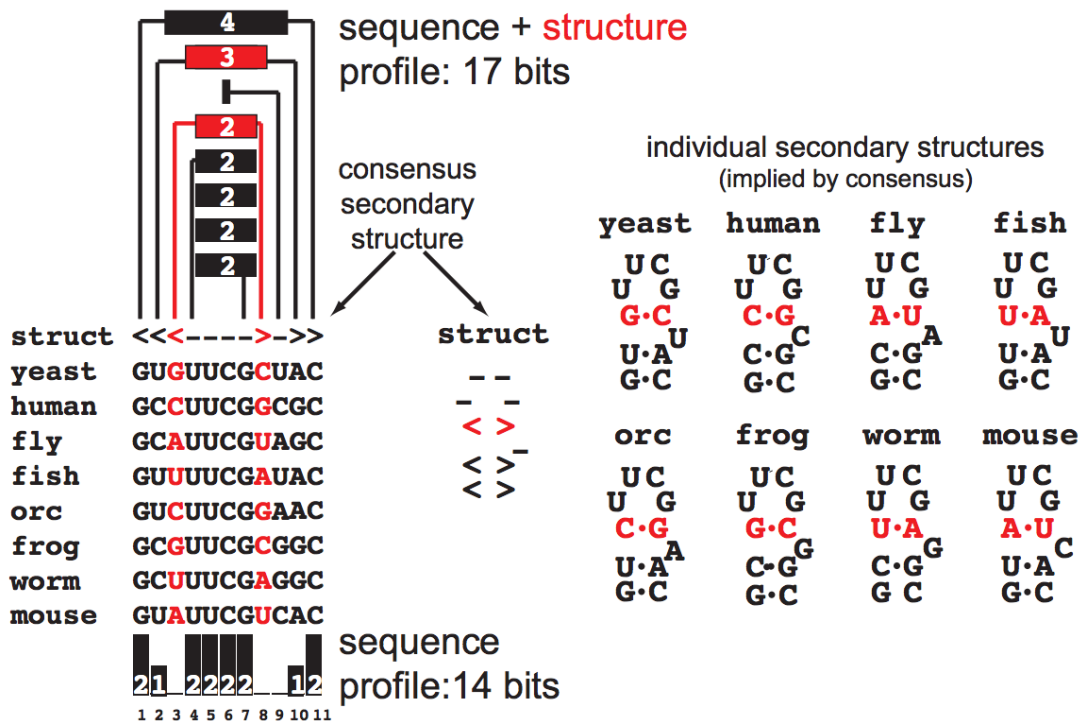
ExAlt – yep, another HMM



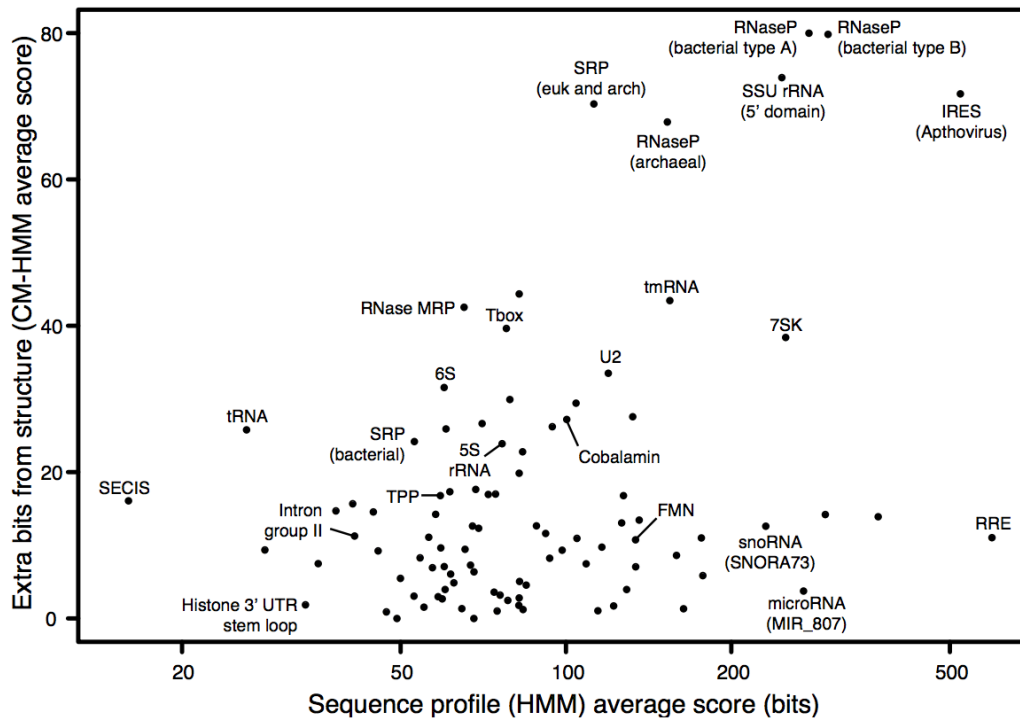
manual (re)annotation tools

- prokaryotic: Manatee/Ergatis, MaGe/MicroScape, ...
- eukaryotic: Apollo, Artemis, ZMAP/Otterlace, ACEdb, ...

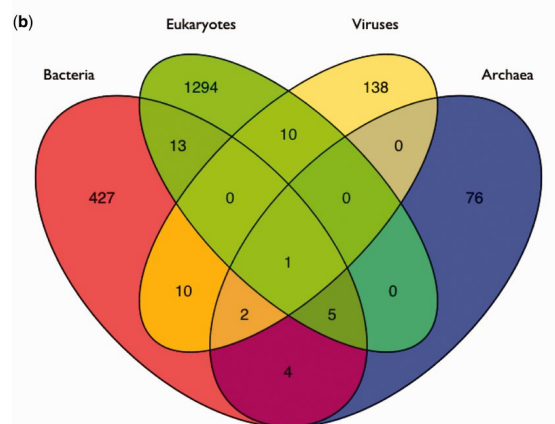
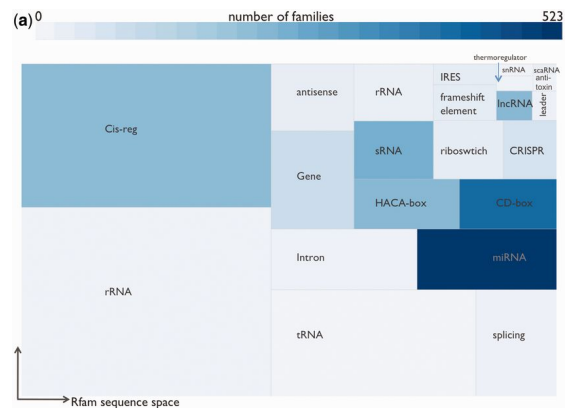
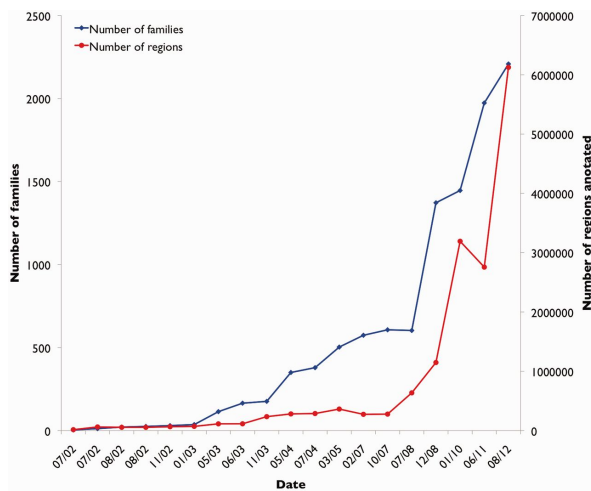
ncRNA gene finding: Infernal



ncRNA gene finding: Infernal



RFAM



recap: eukaryotic gene prediction

- still a hard problem to distinguish genes from genome
- challenges to consider: alternative splicing, pseudogenes, incomplete UTRs, cis/trans-splicing, ...
- ncRNA limited by length, unknown families
- euk annotation pipelines (Ensembl, UCSC, MAKER, etc.) integrate many algorithms
- will RNAseq eliminate our need for such tools? ... tune in Friday