Genome 559
Intro to Statistical and Computational Genomics

Lecture 16a:

Computational Gene Prediction

Larry Ruzzo

Today:

Finding protein-coding genes

  coding sequence statistics

  prokaryotes

  mammals

More on classes

More practice

Codons & The Genetic Code

		Second Base					
		U	C	A	G		
First Base	U	Phe	Ser	Tyr	Cys	U	Third Base
		Phe	Ser	Tyr	Cys	C	
		Leu	Ser	Stop	Stop	A	
		Leu	Ser	Stop	Trp	G	
	C	Leu	Pro	His	Arg	U	
		Leu	Pro	His	Arg	C	
		Leu	Pro	Gln	Arg	A	
		Leu	Pro	Gln	Arg	G	
	A	Ile	Thr	Asn	Ser	U	
		Ile	Thr	Asn	Ser	C	
		Ile	Thr	Lys	Arg	A	
		Met/Start	Thr	Lys	Arg	G	
	G	Val	Ala	Asp	Gly	U	
		Val	Ala	Asp	Gly	C	
		Val	Ala	Glu	Gly	A	
		Val	Ala	Glu	Gly	G	

Ala	: Alanine
Arg	: Arginine
Asn	: Asparagine
Asp	: Aspartic acid
Cys	: Cysteine
Gln	: Glutamine
Glu	: Glutamic acid
Gly	: Glycine
His	: Histidine
Ile	: Isoleucine
Leu	: Leucine
Lys	: Lysine
Met	: Methionine
Phe	: Phenylalanine
Pro	: Proline
Ser	: Serine
Thr	: Threonine
Trp	: Tryptophane
Tyr	: Tyrosine
Val	: Valine

Idea #1: Find Long ORF’s

Reading frame: which of the 3 possible sequences of triples does the ribosome read?

Open Reading Frame: No stop codons

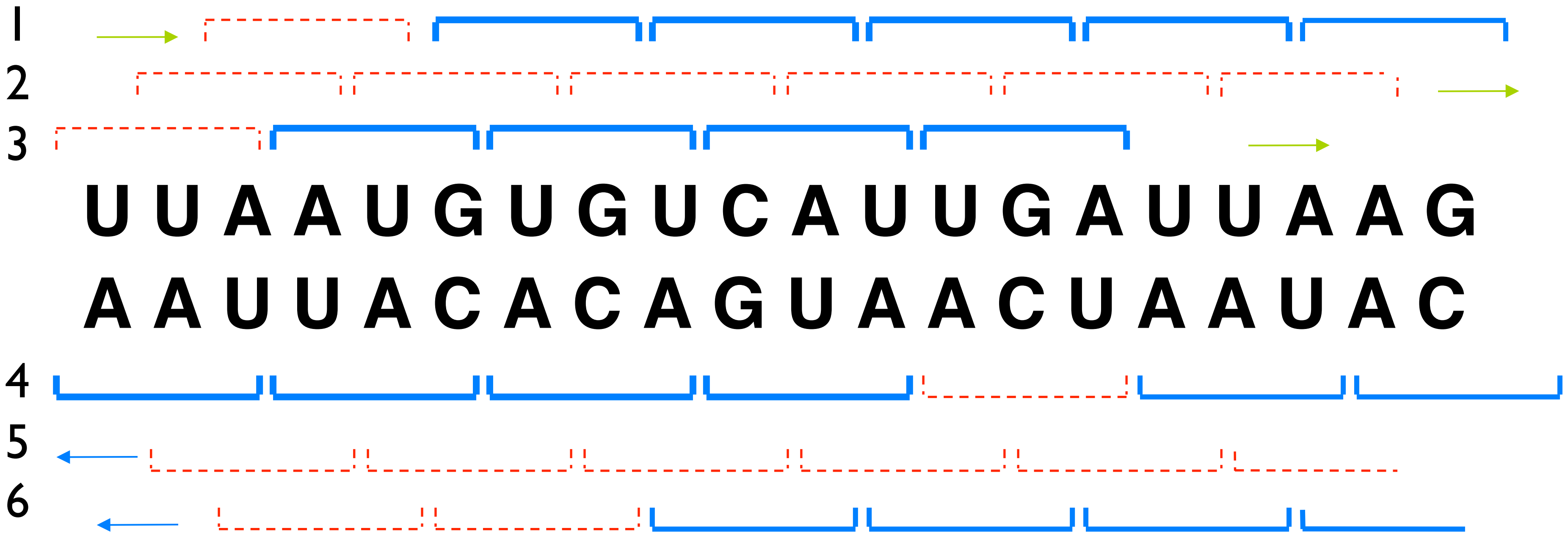
In random DNA

- average ORF = 64/3 = 21 triplets
- 300bp ORF once per 36kbp per strand

But average protein ~ 1000bp

So, coding DNA is not random–stops are rare

Scanning for ORFs



UUAAUGUGUCAUUGAUUAAG

AAUUACACAGUAACUAAUAC

Idea #2: Codon Frequency,…

Even between stops, coding DNA is not random

In random DNA, Leu : Ala : Tryp = 6 : 4 : 1

But in real protein, ratios ~ 6.9 : 6.5 : 1

Even more: *synonym usage* is biased (in a species dependant way)

- Examples known with 90% AT 3rd base
- Why? E.g. efficiency, histone, enhancer, splice interactions,…

More generally: k-th order Markov model

- k=5 or 6 is typical, since significant influences spanning codons are detectable

Markov Models

Can always represent a joint probability distribution

$$P(x_1 x_2 \ldots x_n) = P(x_1)\ P(x_2 \mid x_1)\ P(x_3 \mid x_1 x_2)\ \ldots\ P(x_n \mid x_1 x_2 \ldots x_{n-3} x_{n-2} x_{n-1})$$

If each letter only depends on the k previous ones, it’s a “k-th order Markov model.” E.g., k=3:

$$P(x) = P(x_1)\ P(x_2 \mid x_1)\ P(x_3 \mid x_1 x_2)\ P(x_4 \mid x_1 x_2 x_3)\ P(x_5 \mid x_2 x_3 x_4)\ \ldots\ P(x_n \mid x_{n-3} x_{n-2} x_{n-1})$$

Idea: *distant influences fade*

Implementation: count (k+1)-mers; frequency of k+1st letter conditional on previous k is P(-|-) above.

(It’s MLE; maybe add pseudocounts, too. Sound familiar…?)

For “gene finding”

Given:

P(- | -) for known genes, vs

Q(- | -) for background,

again can look at likelihood ratio

P/Q (or log(P/Q))

that given sequence comes from the “gene” model vs the “background” model.

Overall, “sliding window” ≈ like WMM scoring

Report high scores.

Codon Usage in Φx174

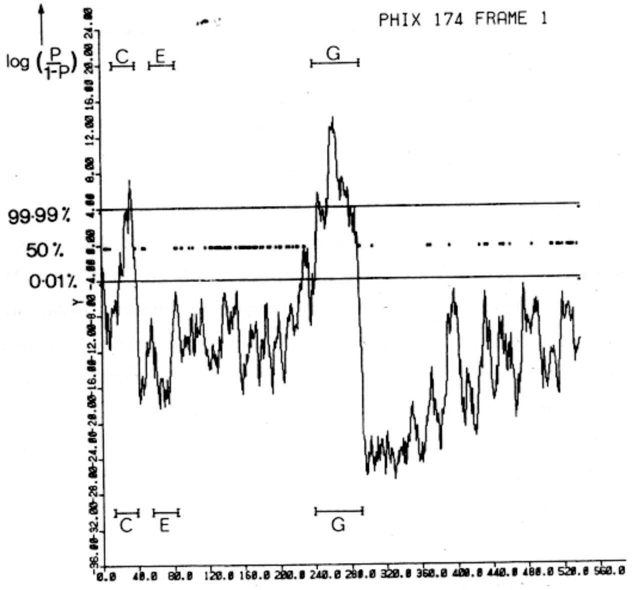


Fig 2a

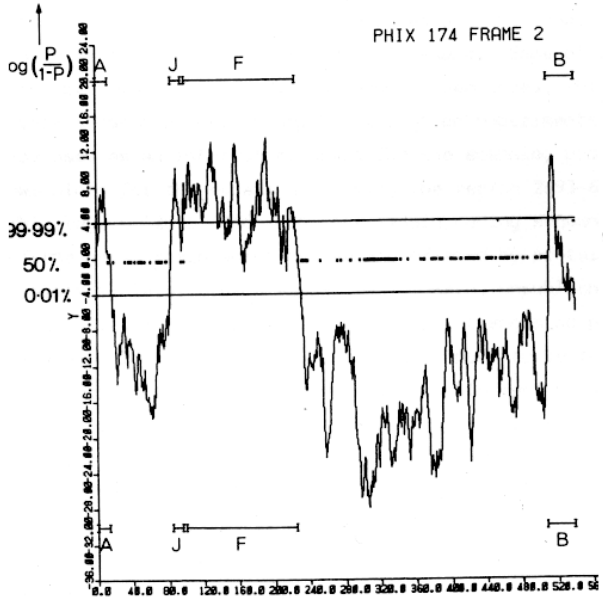


Fig 2b

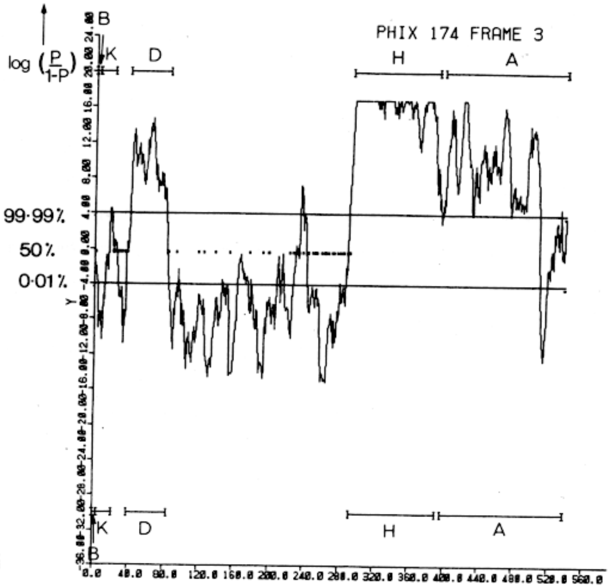


Fig 2c

Staden & McLachlan, NAR 10, 1 1982, 141-156

Summary

Computational gene prediction exploits statistical differences between protein coding genes and other DNA sequence, e.g.

long ORFS

codon-usage- or other baises

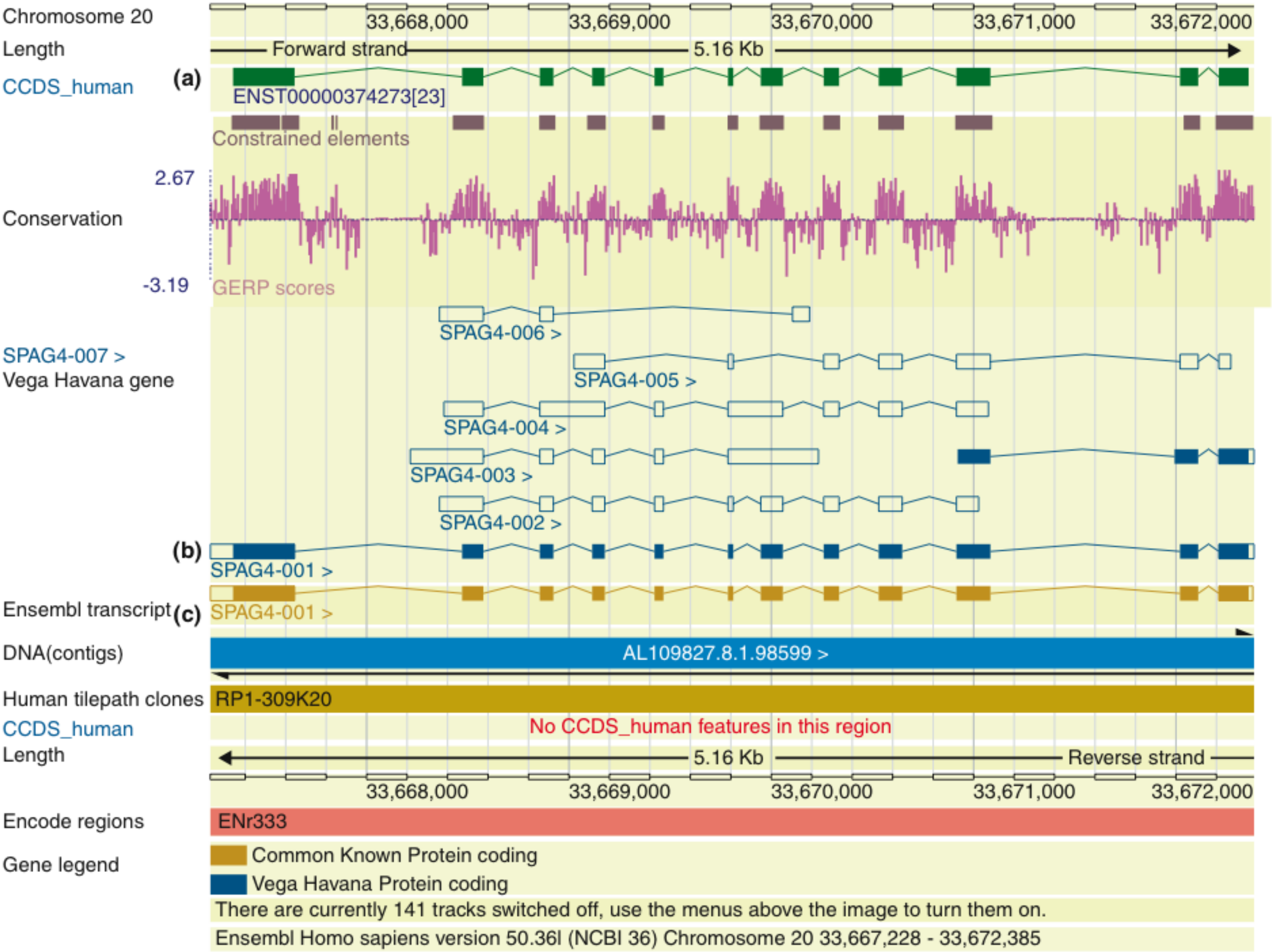
Often use kth-order Markov models, $k \approx 6$

This works pretty well in prokaryotes

Eukaryotes are harder…

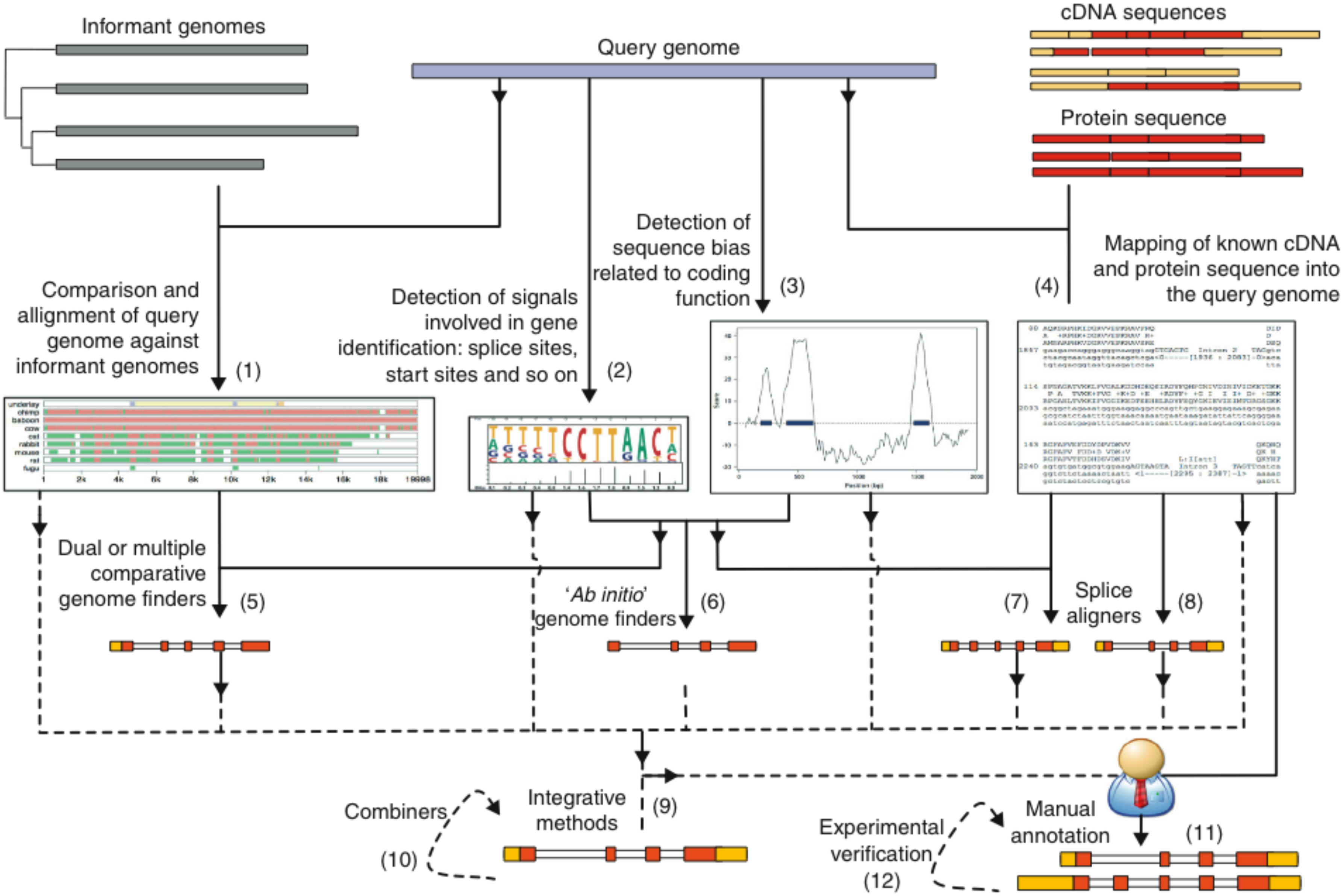
In addition to larger genomes, splicing, alternative splice-, transcription start- and/or, polyA-sites

> “Mammalian transcriptomes are a composed of a swarming mass of different, overlapping transcripts...”
>
> Harrow, *et al.* Identifying protein-coding genes in genomic sequences. Genome Biol. 2009,10(1):201.



There are currently 141 tracks switched off, use the menus above the image to turn them on.

Ensembl Homo sapiens version 50.36l (NCBI 36) Chromosome 20 33,667,228 - 33,672,385


Informant genomes
Query genome
cDNA sequences
Protein sequence
Comparison and allignment of query genome against informant genomes
(1)
Detection of signals involved in gene identification: splice sites, start sites and so on
(2)
Detection of sequence bias related to coding function
(3)
(4)
Mapping of known cDNA and protein sequence into the query genome
Dual or multiple comparative genome finders
(5)
'Ab initio' genome finders
(6)
(7)
Splice aligners
(8)
Combiners
(10)
Integrative methods
(9)
Experimental verification
(12)
Manual annotation
(11)

Summary

Integrate many sources of information

Many tools you've seen:

BLAST, pairwise alignment, multiple alignment, sequence profiles/weight matrix/Markov/phylogenetic modeling

And extensions:

Hidden Markov models, *spliced* alignment, ...

Assessment:

purely computational predictions – ~80% accurate on exons, ~60% on genes (e.g., often extra/missing exons)

So, manual curation still valuable