

**FACULTAD DE INGENIERÍA Y CIENCIAS EXACTAS
DEPARTAMENTO DE BIOTECNOLOGÍA Y TECNOLOGÍA ALIMENTARIA
UNIVERSIDAD ARGENTINA DE LA EMPRESA**

Bioinformática

ANÁLISIS COMPUTACIONAL DE SECUENCIAS

Dr. Lucas L. Maldonado (PhD)

Lic. Biotechnologist and Molecular Biologist

Bioinformatics and genomics specialist

CONICET

Fac. de Medicina - UBA

Fac. de Ciencias Exactas y Naturales – UBA

lucamaldonado@uade.edu.ar

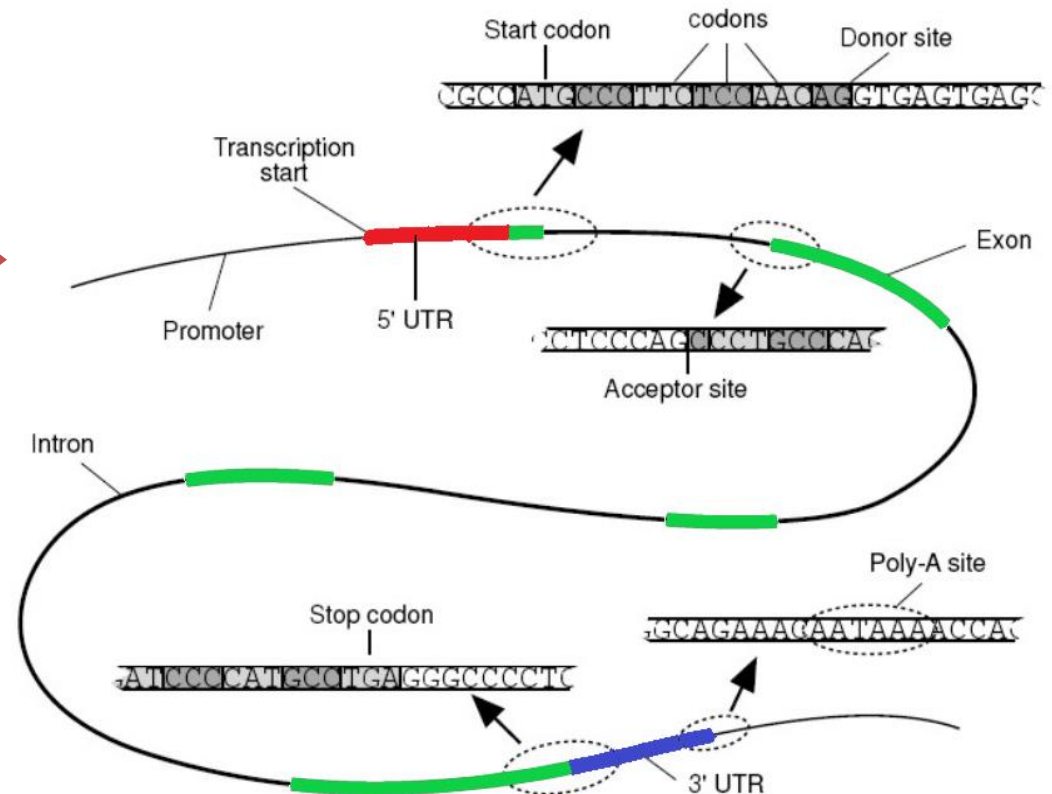
lmaldonado@fmed.uba.ar

lucas.l.maldonado@gmail.com.ar

Genomic annotation: from sequence to predicted function

Raw sequence data: millions and millions of nucleotides

```
AAACACTTAGACAATCAATATAAAGATGAAGTGAACGC
TCTTAAAGAGAAGTTGGAAAACCTGCAGGAACAAATCA
AAGATCAAAAAAGGATAGAAGAACAAGAAAAACCACAA
ACACTTAGACAATCAATATAAAGATGAAGTGAACGCTC
TTAAAGAGAAGTTGGAAAACCTGCAGGAACAAATCAAA
GATCAAAAAAGGATAGAAGAACAAGAAAAACCACAAAC
ACTTAGACAATCAATATAAAGATGAAGTGAACGCTCTT
AAAGAGAAGTTGGAAAACCTGCAGGAACAAATCAAAGA
TCAAAAAAGGATAGAAGAACAAGAAAAACCACAAACAC
TTAGACAATCAATATAAAGATGAAGTGAACGCTCTTAA
AGAGAAGTTGGAAAACCTGCAGGAACAAATCAAAGATC
AAAAAAGGATAGAAGAACAAGAAAAACCACAAACACTT
AGACAATCAATATAAAGATGAAGTGAACGCTCTTAAAG
AGAAGTTGGAAAACCTGCAGGAACAAATCAAAGATCAA
AAAAGGATAGAAGAACAAGAAAAACCACAAACACTTAG
ACAATCAATATAAAGATGAAGTGAACGCTCTTAAAGAG
AAGTTGGAAAACCTGCAGGAACAAATCAAAGATCAAAA
AAGGATAGAAGAACAAGAAAAACCAC
```

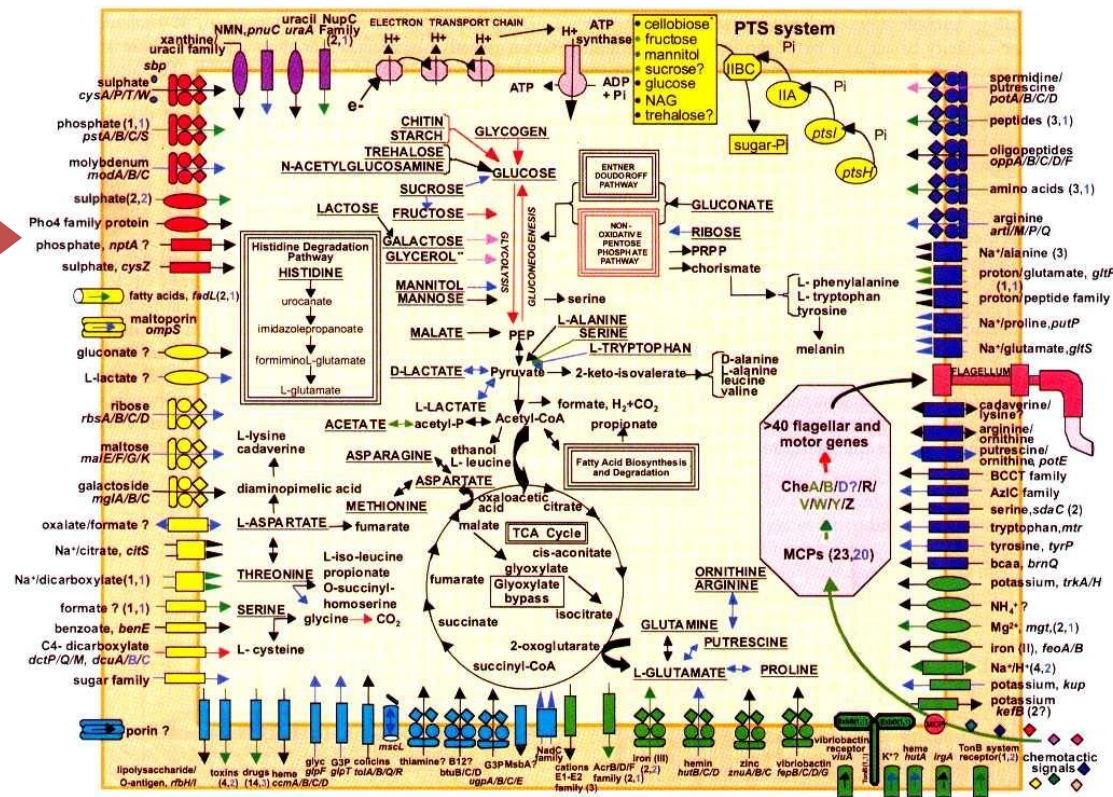


Genomic annotation: from sequence to predicted function

A virtual cell:
overview of predicted pathways

Raw sequence data: millions
and millions of nucleotides

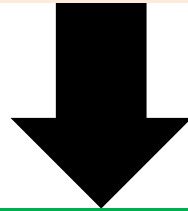
AAACACTTAGACAATCAATATAAAGATGAAGTGAACGC
TCTTAAAGAGAAGTTGGAAAACCTGCAGGAACAAATCA
AAGATCAAAAAGGATAGAAGAACAAGAAAACCACAA
ACACTTAGACAATCAATATAAAGATGAAGTGAACGCTC
TTAAAGAGAAGTTGGAAAACCTGCAGGAACAAATCAA
GATCAAAAAGGATAGAAGAACAAGAAAACCACAAAC
ACTTAGACAATCAATATAAAGATGAAGTGAACGCTCTT
AAAGAGAAGTTGGAAAACCTGCAGGAACAAATCAAAG
TCAAAAAGGATAGAAGAACAAGAAAACCACAAACAC
TTAGACAATCAATATAAAGATGAAGTGAACGCTCTTAA
AGAGAAGTTGGAAAACCTGCAGGAACAAATCAAAGATC
AAAAAGGATAGAAGAACAAGAAAACCACAAACACTT
AGACAATCAATATAAAGATGAAGTGAACGCTCTTAAAG
AGAAGTTGGAAAACCTGCAGGAACAAATCAAAGATCAA
AAAAGGATAGAAGAACAAGAAAACCACAAACACTTAG
ACAATCAATATAAAGATGAAGTGAACGCTCTTAAAGAG
AAGTTGGAAAACCTGCAGGAACAAATCAAAGATCAAAA
AAGGATAGAAGAACAAGAAAACCAC



Genome annotation

Structural gene annotation

Structural genome annotation is the process of identifying the exact location of **genes** and all of the coding and non-coding regions in a **genome** and determining intron, exons and other elements



Functional gene annotation

Functional gene annotation means the description of the biochemical and biological function of proteins. Gene products, proteins and the description of protein domains as well as assigning Gene Ontology Annotations (GO terms)

Structural and Functional Annotation

Structural annotation

Identification of genomic elements.

- ORFs predicted during genome assembly
- Location of ORFs
- Gene structure
- Coding regions
- Location of regulatory motifs etc

Functional annotation

Attaching biological information to genomic elements.

- Biochemical function
- Biological function
- Involved regulation and interactions
- Expression etc

These steps may involve both biological experiments and *in silico* analysis.

Composición del genoma

1. Genes

- mRNA/proteínas – 5'UTR/3'UTR
- RNAs
- miRNAs
- tRNAs
- ncRNA

2. Pseudogenes

3. Regiones no codificantes

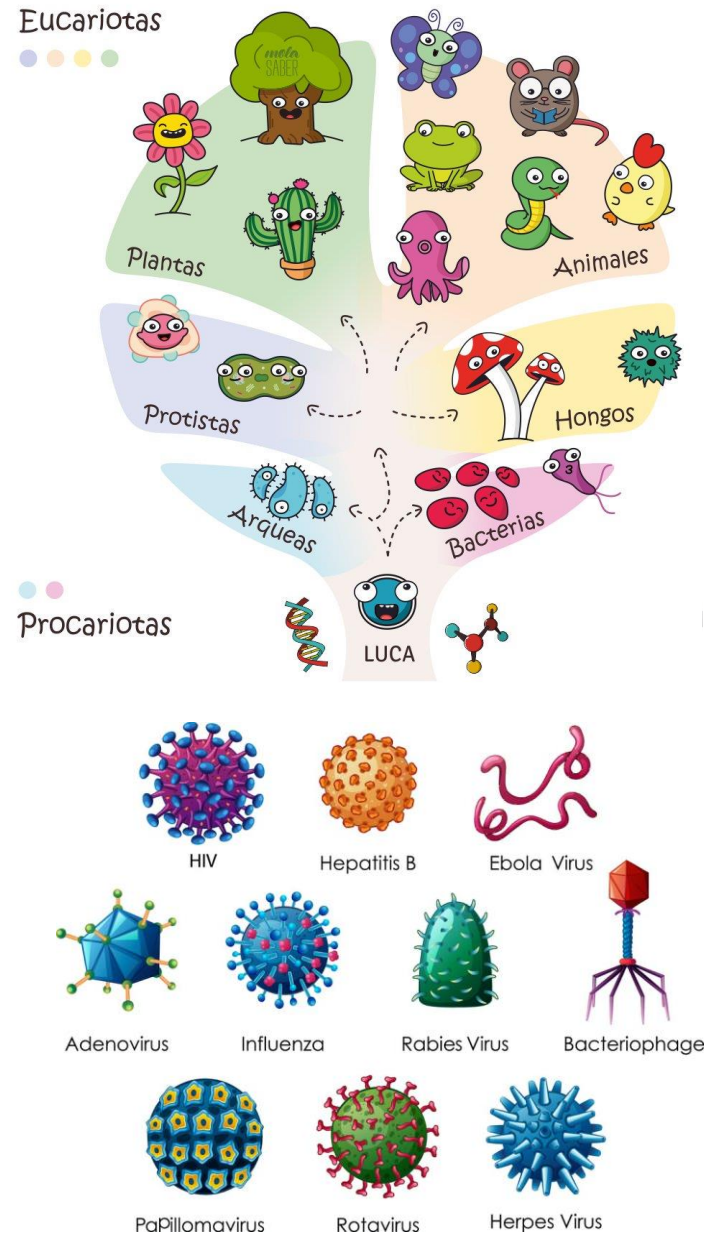
- Transposones
- SINEs/LINEs
- Regulatorias
- Estructurales

1. Estructura de cromatina

2. Mitosis / meiosis

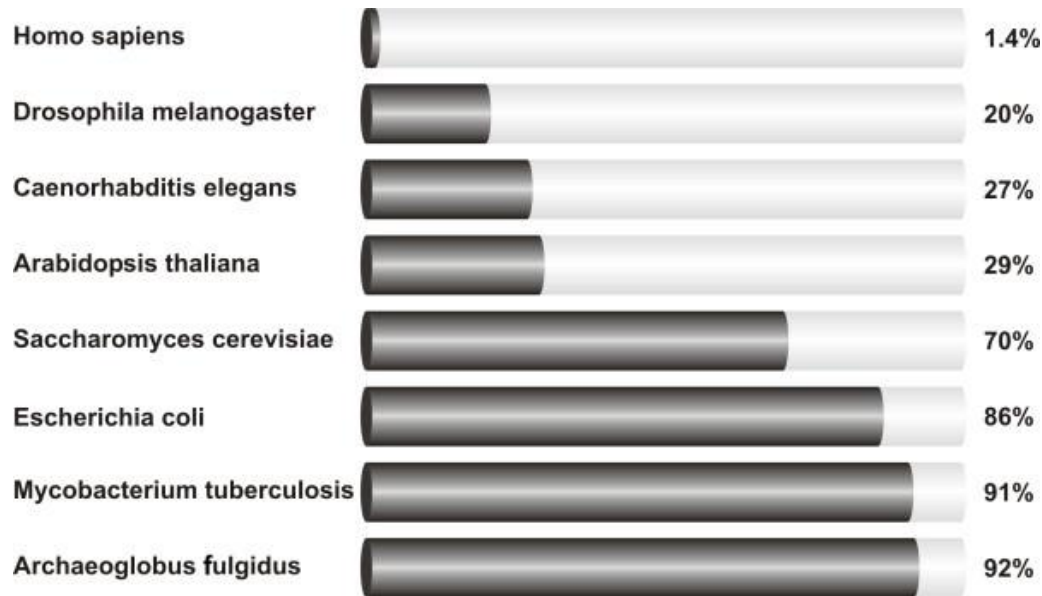
3. Repeticiones

Reinos de la vida



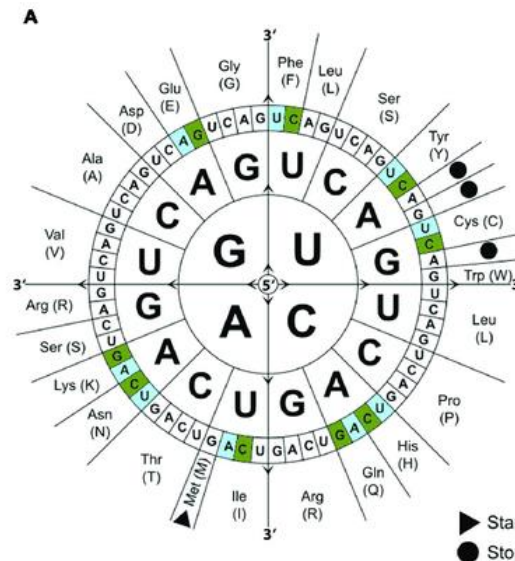
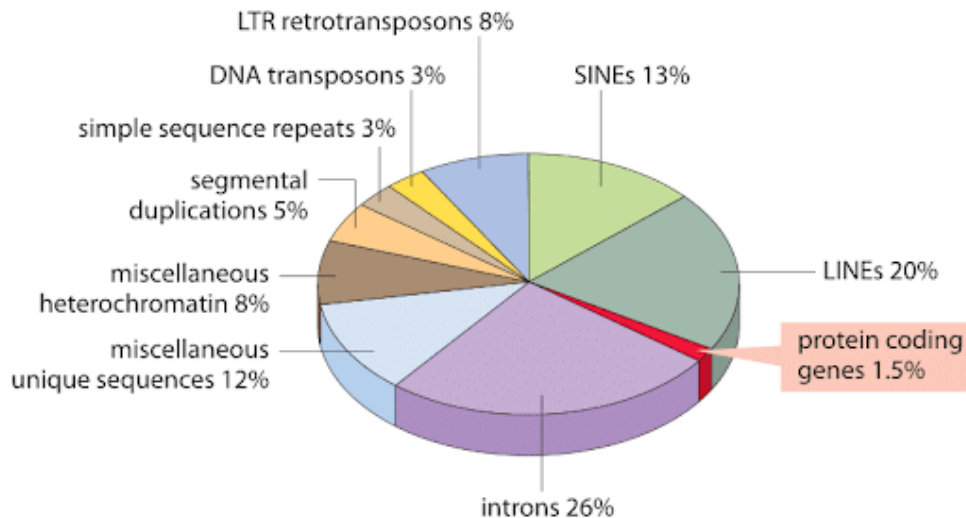
Composición del genoma

Coding genes



!= organism equal to != codon usage
Different codon usage tables

main components of the human genome



B

Amino acid	Codon	<i>P. patens</i>	<i>A. thaliana</i>	<i>S. cerevisiae</i>	<i>S. pombe</i>	<i>H. sapiens</i>
Cys	UGC	+	+	+	+	+
	UGU	-	-	+	-	-
Glu	GAA	-	-	+	-	-
	GAG	+	+	+	+	+
Phe	UUC	+	+	+	+	+
	UUU	-	-	-	-	-
His	CAC	+	+	+	+	+
	CAU	-	-	-	-	-
Ile	AUA	-	-	/	-	-
	AUC	+	+	/	-	-
	AUU	-	-	-	-	-
Lys	AAA	-	-	-	-	-
	AAG	+	+	+	+	+
Asn	AAC	+	+	+	+	+
	AAU	-	-	-	-	-
Gln	CAA	-	-	-	+	-
	CAG	+	+	+	+	+
Tyr	UAC	+	+	+	+	+
	UAU	-	-	-	-	-

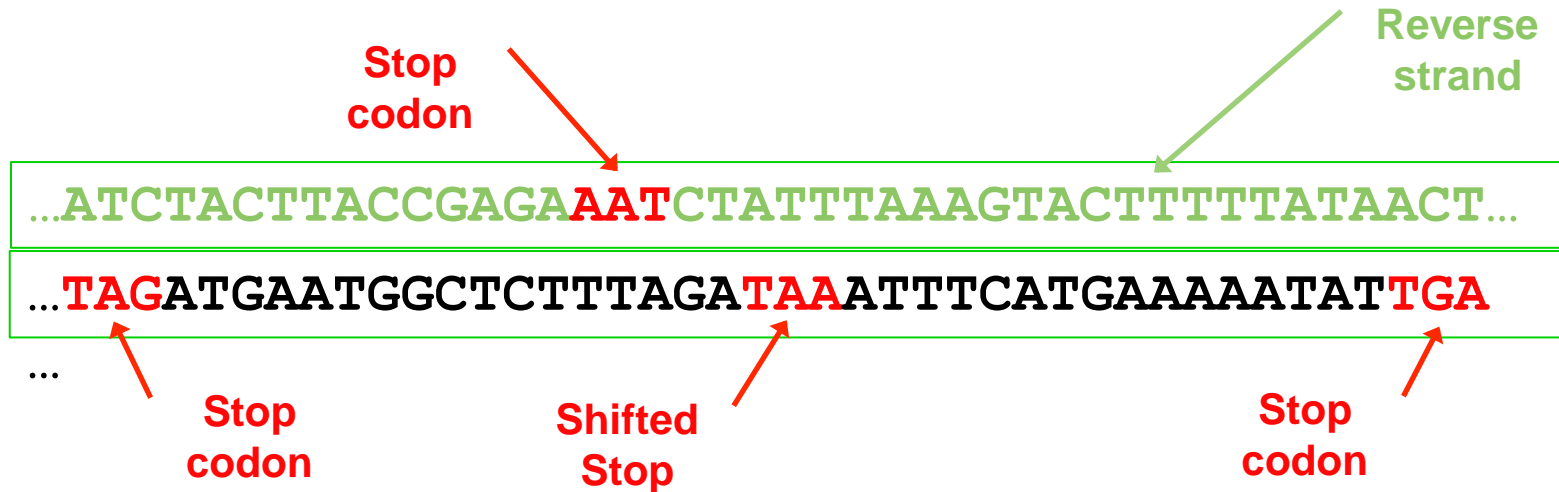
Bacterias

- Find open reading frames (ORFs).



Bacteria

- Find open reading frames (ORFs).



- But ORFs generally overlap ...

Bacterias

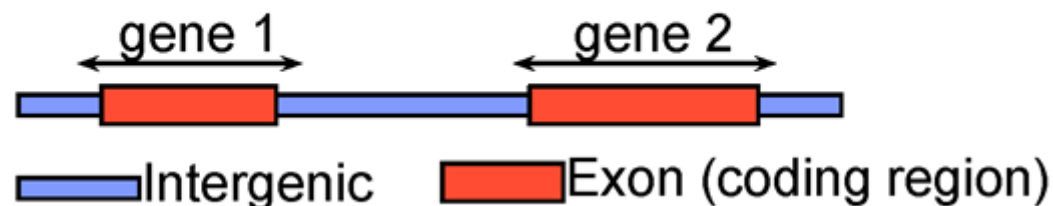
Gene Prediction: Computational Challenge

aatgcatgcggtatgctaataatgcatgcggtatgctaagctgggatccgatgacaatgcatgcggtatgctaa
tgcataatggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggtatgctaa
atggtcttgggatttaccttggaatgctaataatgcatgcggtatgctaagctgggatccgatgacaatgcatg
ggctatgctaataatgcatgcggtatgcaagctgggatccgatgactatgctaagctgcggtatgctaataatgcatg
cggtatgctaagctgggatccgatgacaatgcatgcggtatgctaataatgcatgcggtatgcaagctgggatc
ctgcggtatgctaataatggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcg
ctatgctaataatggtcttgggatttaccttggaatgctaataatgcatgcggtatgctaagctgggaatgcatg
cggtatgctaagctgggatccgatgacaatgcatgcggtatgctaataatgcatgcggtatgcaagctgggatc
cgatgactatgctaagctgcggtatgctaataatgcatgcggtatgctaagctcatgcggtatgctaagctggg
aatgcatgcggtatgctaagctgggatccgatgacaatgcatgcggtatgctaataatgcatgcggtatgcaag
ctgggatccgatgactatgctaagctgcggtatgctaataatgcatgcggtatgctaagctcggctatgctaata
atggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggtatgctaataatggtc
ttgggatttaccttggaatgctaataatgcatgcggtatgctaagctgggaatgcatgcggtatgctaagctgg
gatccgatgacaatgcatgcggtatgctaataatgcatgcggtatgcaagctgggatccgatgactatgctaagc
tgcggtatgctaataatgcatgcggtatgctaagctcatgcggtatgctaagctgggaatgcatgcggtatgctaagctgg

Bacteria

Gene Prediction: Computational Challenge

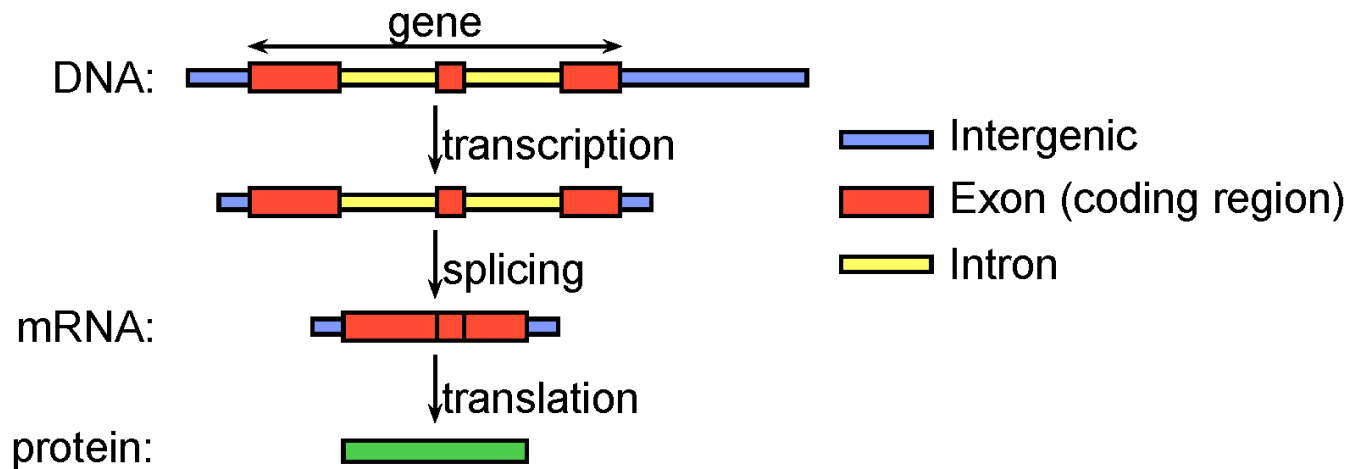
aatgcatgcggctatgctaatgcatgcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaa
tgcatgcggctatgcaagctgggatccgatgactatgctaagctgggatccgatgacaatgcatgcggctatgc
taatgaatggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggctatgctaatga
atggtcttgggatttaccttggaatgctaatgcatgcggctatgctaagctgggatccgatgacaatgcatgc
ggctatgctaatgcatgcggctatgcaagctgggatccgatgactatgctaagctgcggctatgctaatgcatg
cggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatc
ctgcggctatgctaatgaatggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcgg
ctatgctaatgaatggtcttgggatttaccttggaatgctaatgcatgcggctatgctaagctgggaatgcatg
cggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatc
cgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaagctcatgcggctatgctaagctggg
aatgcatgcggctatgctaagctgggatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaag
ctgggatccgatgactatgctaagctgcggctatgctaatgcatgcggctatgctaagctcggctatgctaatga
atggtcttgggatttaccttggaatgctaagctgggatccgatgacaatgcatgcggctatgctaatgaatggtc
ttgggatttaccttggaatgctaatgcatgcggctatgctaagctgggaatgcatgcggctatgctaagctgg
gatccgatgacaatgcatgcggctatgctaatgcatgcggctatgcaagctgggatccgatgactatgctaagc
tgcggctatgc



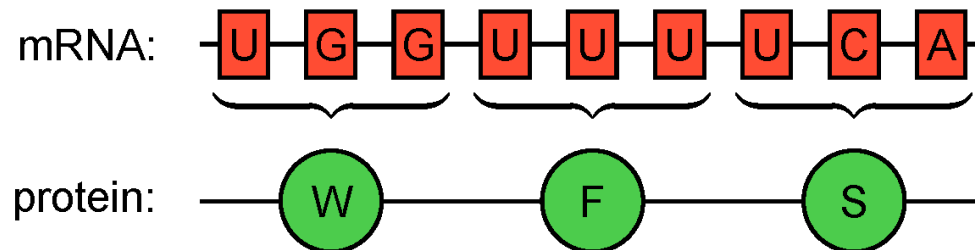
Eukaryotes

Closer look on genes

Process of protein production:



Translation: triple of letters in mRNA → one amino acid in protein



Eukaryotes

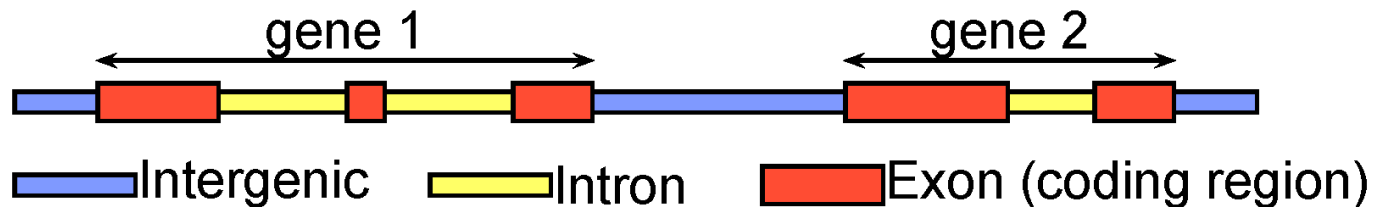
Task of gene finding

cgggtgaaactgcacgattggttgctggcttaaagatagaccaatcagagtggtgtaacgtca
tatttagcgtcttctatcatccaatcactgcactttacacactataaatagagcagctca
tgggcgtatTTTgcgctagtgttgggtgttccgctgtgctgtttttccgtcatggctcgca
ctaagcaaactgctcggaagtctactggtggcaaggcgccacgcaaacagttggccacta
aggcagcccgcaaaagcgctccggccaccggcgggcgtgaaaaagccccaccgctaccggc
cgggcaccgtggctctgcgcgagatccgccgttatcagaagtccactgaactgcttattc
gtaaactacctttccagcgcttgggtgcgcgagattgcgcgaggactttaaaacagacctgc
gtttccagagctccgctgtgatggctctgcaggaggcgtgcgaggcctacttggtagggc
tatttgaggacactaacctgtgcgccatccacgccaagcgcgctcactatcatgcccaagg
acatccagctcgcccgccgcatccgcggagagagggcgtgattactgtggtctctctgac

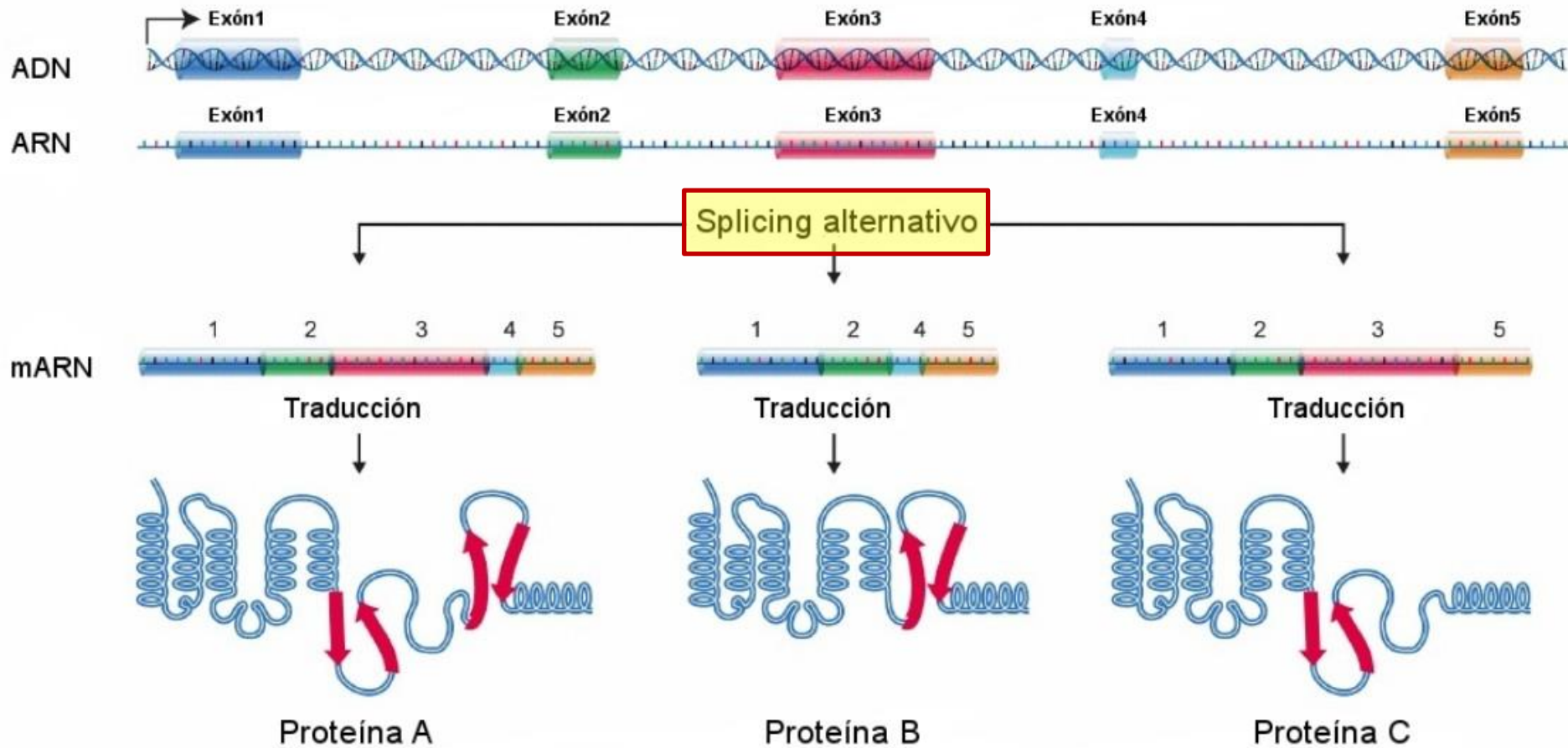
Eukaryotes

Task of gene finding

cggtgaaactgcacgattggtgctggcttaaagatagaccaatcagagtgtgtaacgtca
tatttagcgtcttctatcatccaatcactgcactttacacactataaatagagcagctca
tgggcgtatttgcgctagtgttgggtgttccgctgtgctgtttttccgctc**atggctcgca**
ctaagcaaactgctcggaagtctactggtggcaaggcgccacgcaaacagttggccacta
aggcag**cccgcaaaagcgctccggccaccggcggcgtgaaaaagccccaccgctaccggc**
cgggcaccgtggctctgcgcgagatccgccgttatcagaagtcactgaactgcttattc
gtaaactacctttccagcgctggtg**gcgcgagattgcgcaggactttaaaacag**acctgc
gtttccagagctccgctgtgatggctctgcaggaggcgtgcgaggcctacttggtagggc
tatttgaggacactaacctgtgcgccatccacgccaaagcgcgctcactatcatgcccaagg
acatccagctcgcccgccgcgcatccgcggagagagggcgctga**ttactgtggtctctctgac**



Complejidad del gen eucariota



Procariotas vs Eucariotas

Procariotas:

- Genomas pequeños
- Ausencia de intrones
- Alta densidad de genes
- ORFs solapados
- Genes cortos

Eucariotas:

- Genomas grandes
- Presencia de intrones
- Baja densidad de genes

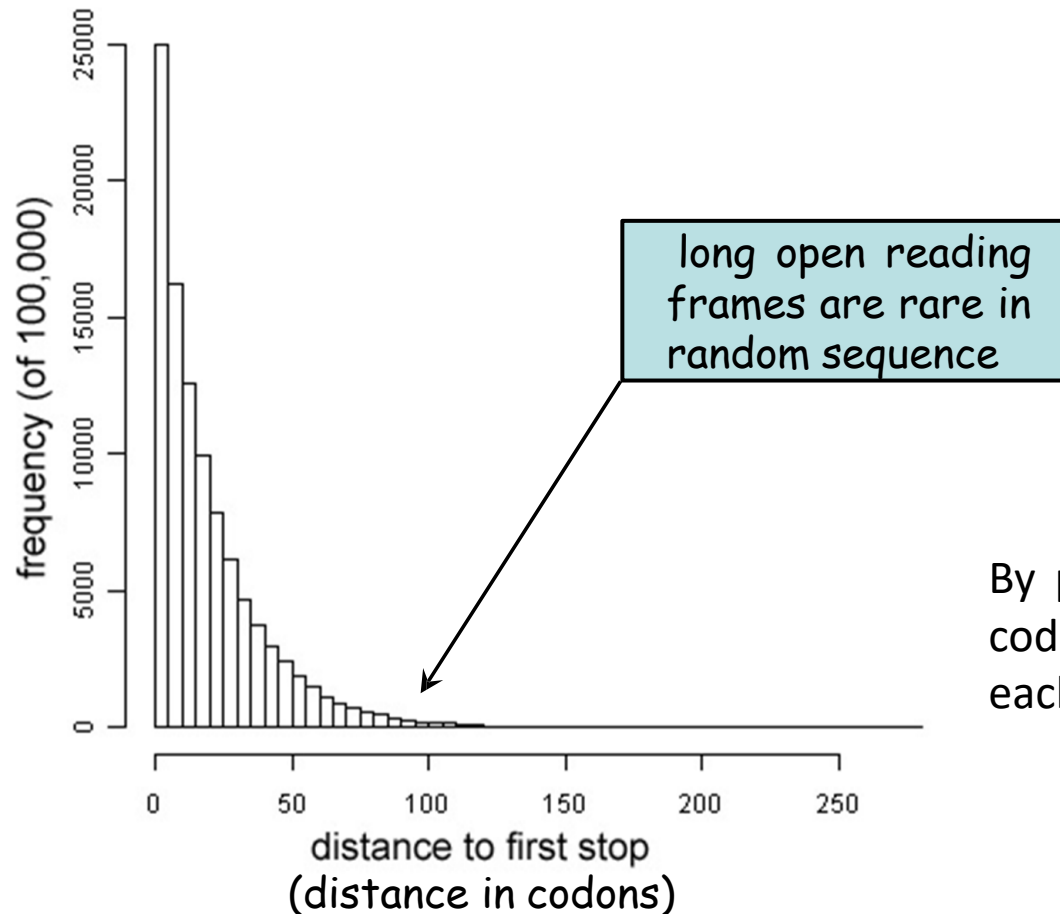
En consecuencia, la predicción de genes, cualquiera sea la estrategia a utilizar es mas simple en procariotas que en eucariotas.

Open reading frames are rarely located randomly in the sequence

- 61 of 64 codons are **not** stop codons (0.953 assuming equal nucleotide frequencies).
- Probability of **not** having a stop codon in a particular reading frame along a length **L** of DNA is a geometric distribution that decays rapidly.
- There are 3 reading frames on each DNA strand.

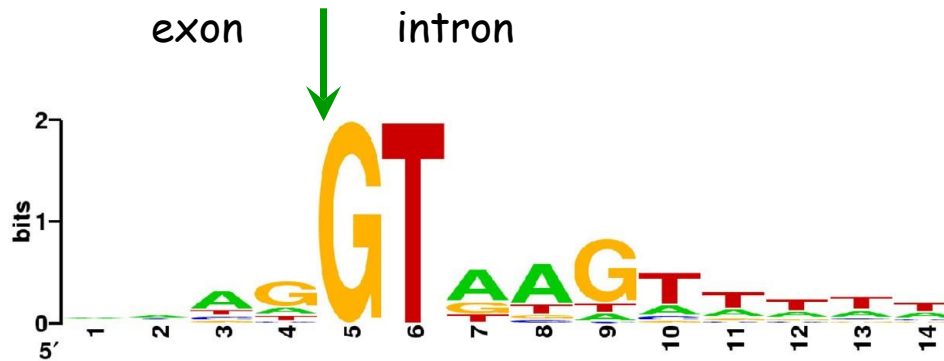
Open reading frames are rarely located randomly in the sequence

Geometric distribution in random sequence of distance to first stop codon ($p=3/64$)

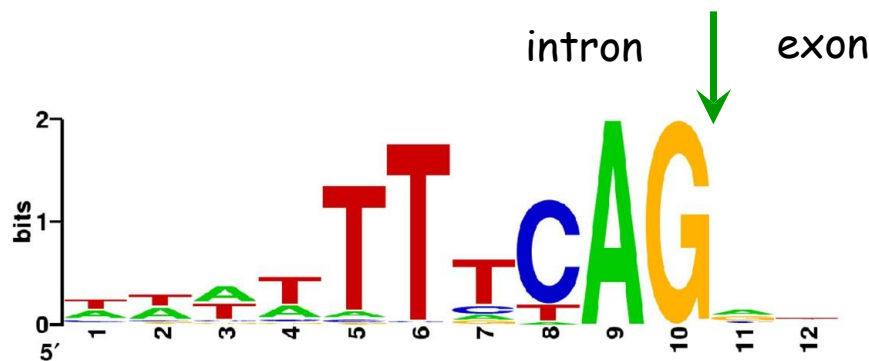


By probability a stop codon should appear each 100 bp

Splice donor and acceptor information



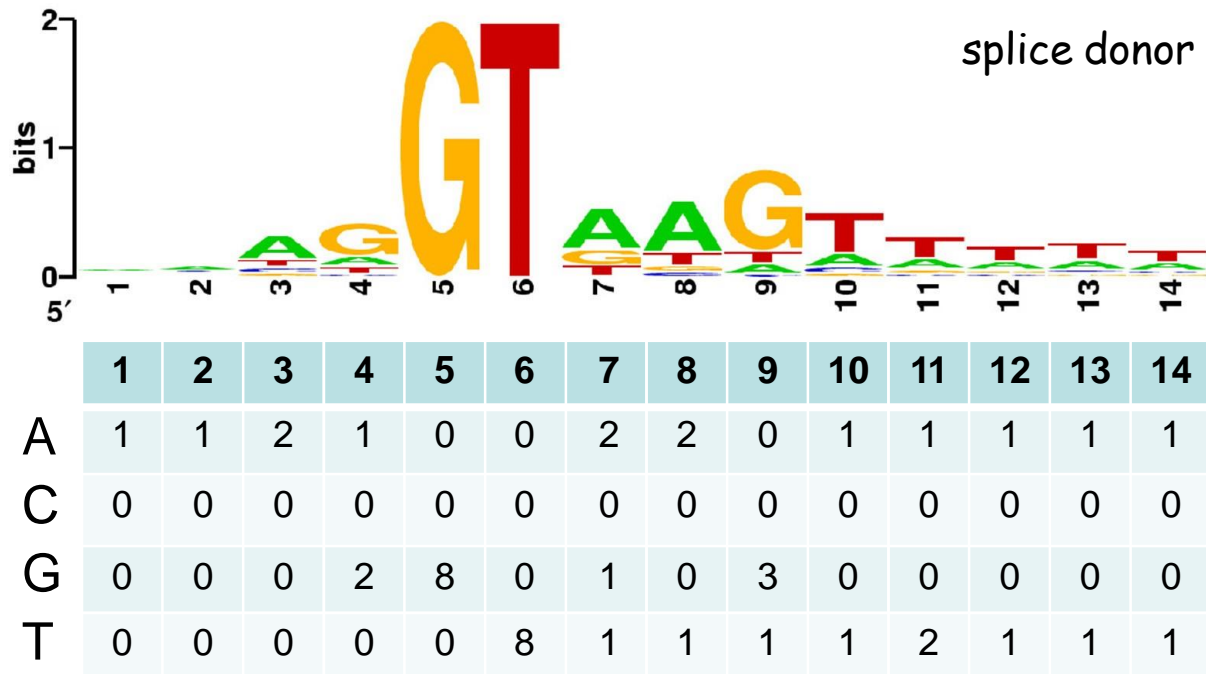
donor, *C. elegans*
(sums to ~8 bits)



acceptor, *C. elegans*
(sums to ~9 bits)

Note - these show a log-odds measure of information content compared to background nucleotide frequencies. Similar to BLOSUM matrix log-odds.

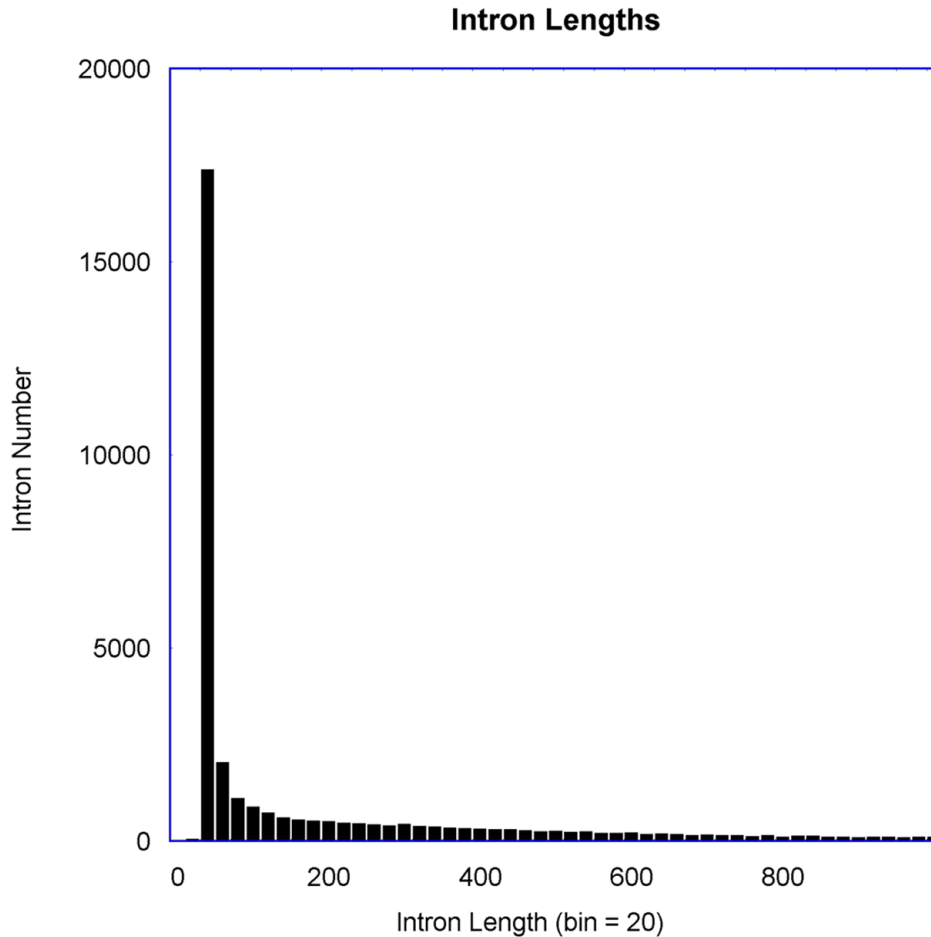
Position Specific Score Matrix (PSSM)



Slide PSSM along DNA, computing a score at every position.

(this is a conceptual example, the real thing would be computed as log-odds values, similar to BLOSUM matrices)

Intron length distribution (C. elegans)



Note: intron length distributions in *Drosophila melanogaster* and *Homo sapiens* (and most other species) are longer and broader.

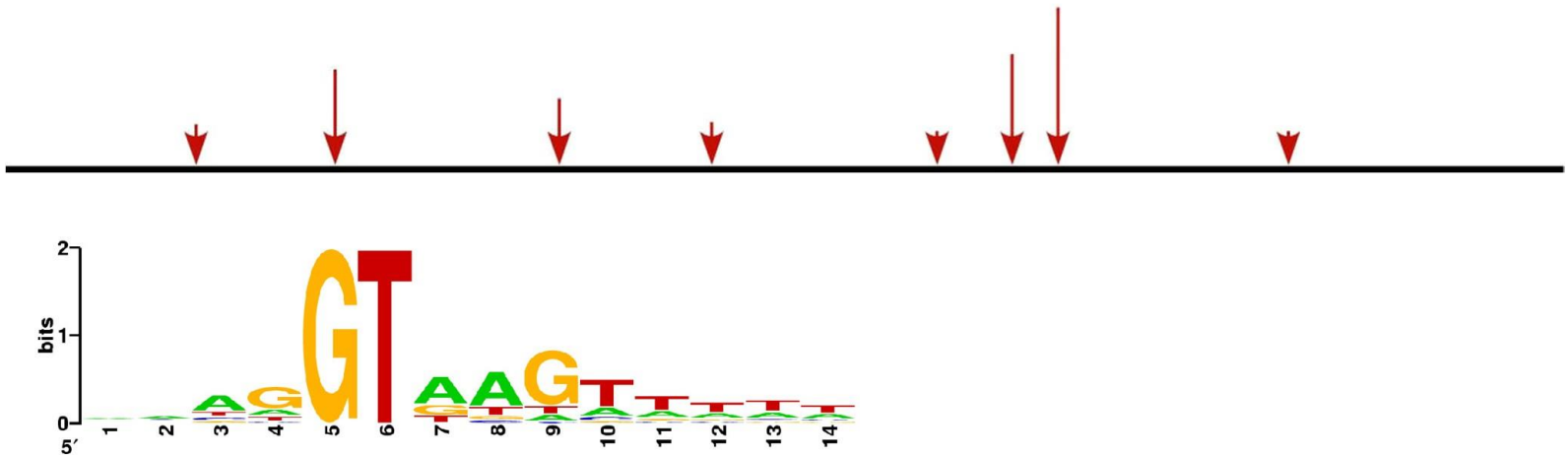
Each species has a characteristic intron length distribution

Other information that can be used

- Splice donor and acceptor must be paired
- Donor must be upstream of acceptor (duh).
- Introns in coding regions must maintain reading frame of the flanking exons.
- Nucleotide content analysis (e.g. introns tend to be **AT** rich).

Simple conceptual example

splice donor candidates (plus strand only)

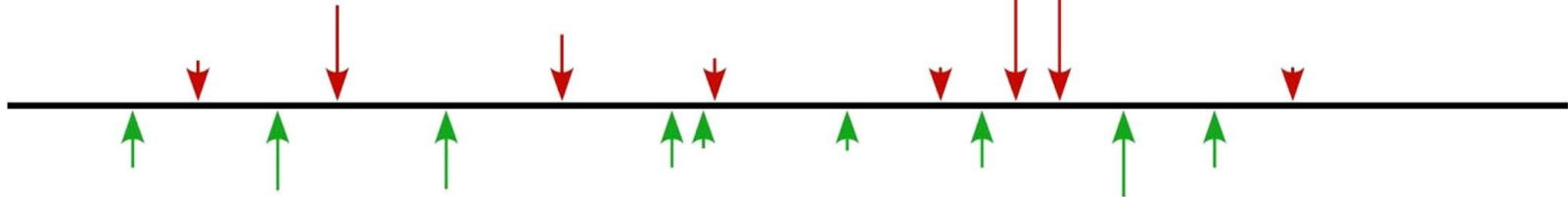


- Sites scored on basis of PSSM matches to known splice donor model (schematized below).
- Arrow length reflects quality of match (worse matches not shown).

Add splice acceptor information

(example cont.)

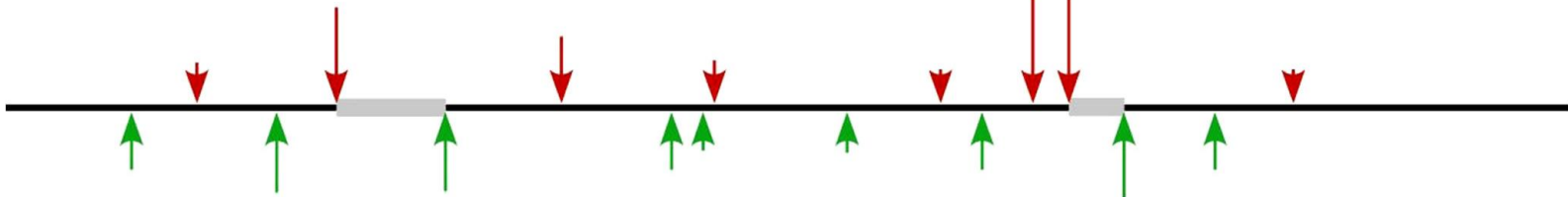
splice donor candidates



splice acceptor candidates

Where would you infer introns?

splice donor candidates

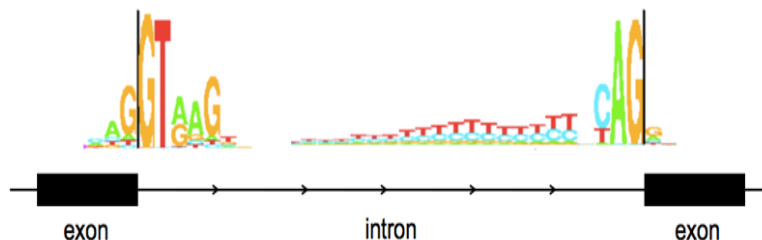


splice acceptor candidates

— = introns (one probable interpretation)

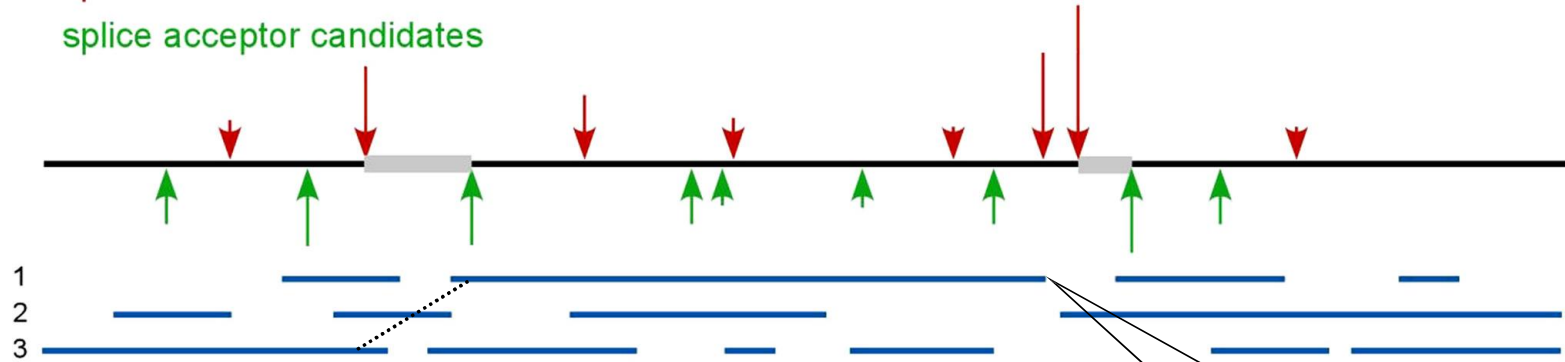
donor

acceptor



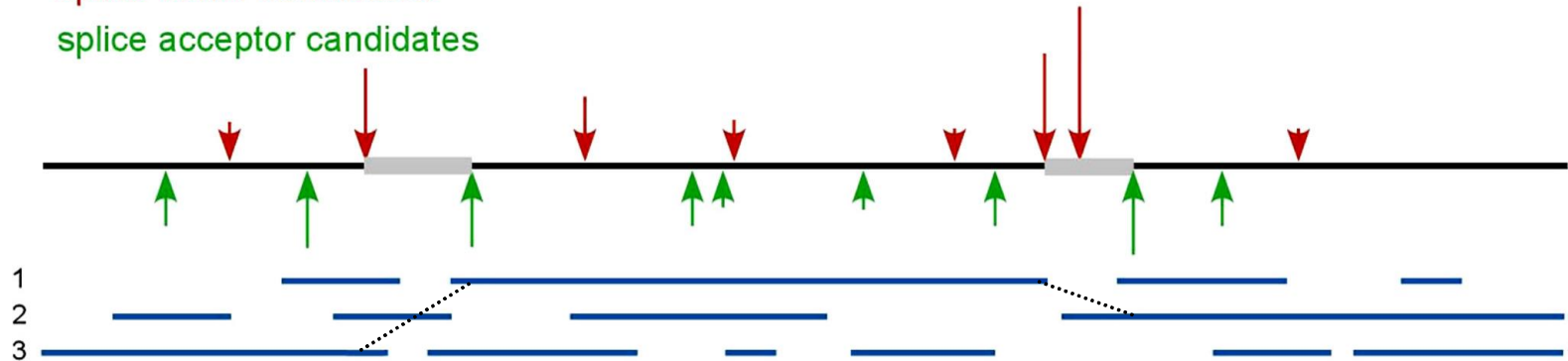
(example cont.)

splice donor candidates
splice acceptor candidates



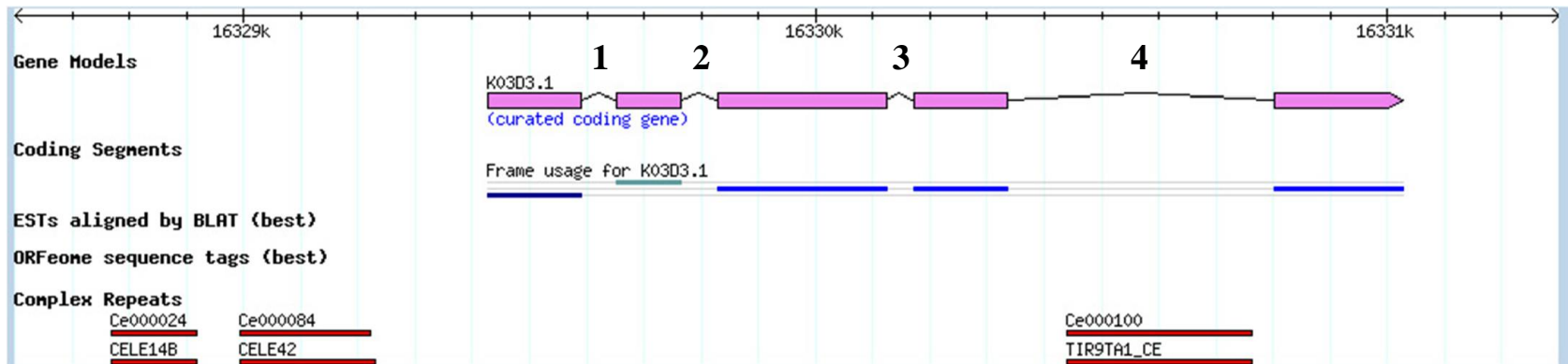
reinterpreted (avoids stop codon by using lower scoring splice donor):

splice donor candidates
splice acceptor candidates



open reading frames (above threshold length, plus strand)

Real example (end result)



Note that this gene has no mRNA sequences (EST and ORFeome tracks empty). This is a pure *ab initio* prediction.

Anotación estructural

Anotación manual:

Expertos usando conocimientos biológicos y bioquímicos

Características de secuencias

Fiable y exacto

Lento y laborioso

Anotación automática:

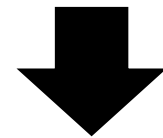
NGS y generación de datos

Herramientas computacionales

Menos exacto

Rápido

Genome project



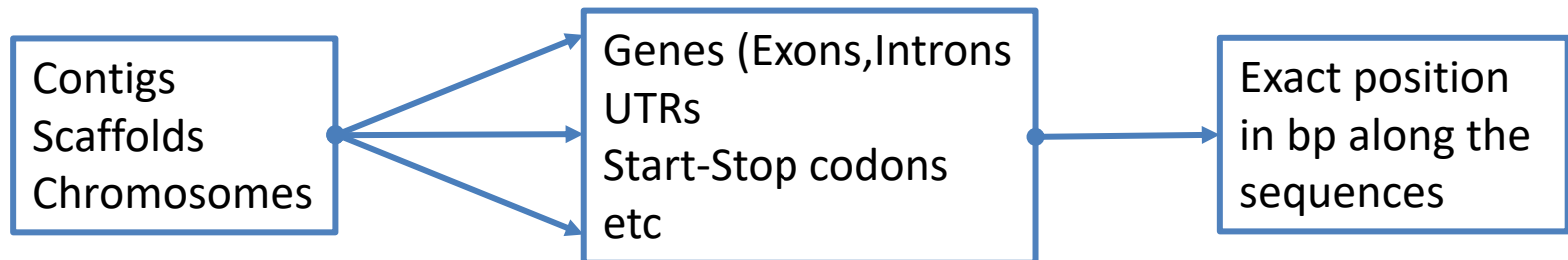
Too many genes



Genome structure annotation

What is it about?

- ❑ Construct a model of the genome
- ❑ Find regions with biological relevance
- ❑ Find coordinates related to biological elements



Genome structure annotation

¿Cómo sería un algoritmo de anotación génica estructural?

1. Screening de toda la secuencia genómica
2. Búsqueda de ORF en la hebra forward y reverse
3. Clasificación de los ORF
4. Selección de los ORF mas probables (ORF mas largos)
5. Identificación de estructuras genómicas típicas

Se busca un codón de inicio a lo largo de toda la secuencia en cada hebra de ADN en los 3 marcos de lectura posibles y se extiende hasta encontrar un codón stop y se seleccionan los mas largos.

What to do before annotation?

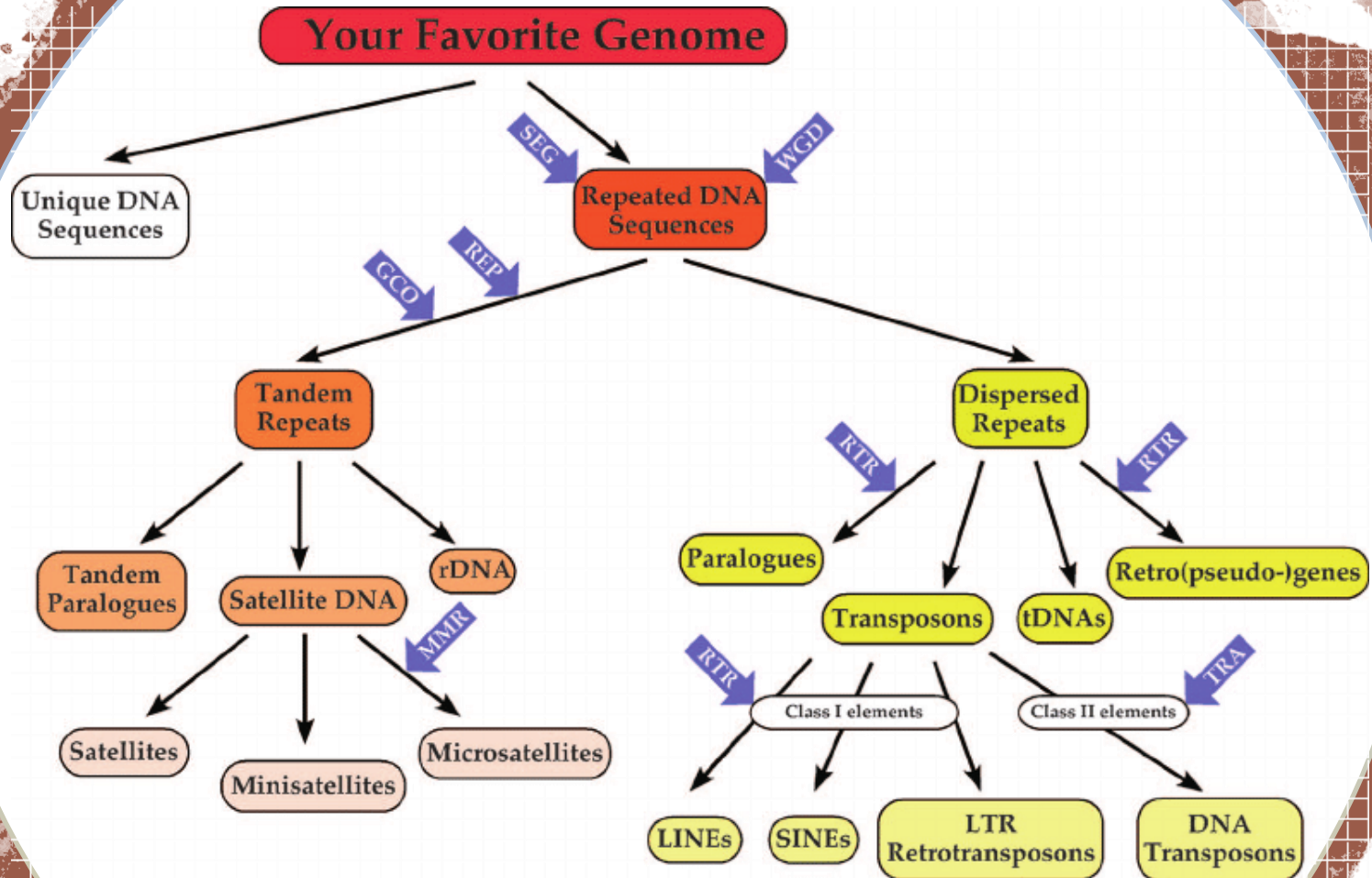
Again annoying repeats:

Essential step before annotation.... Why?

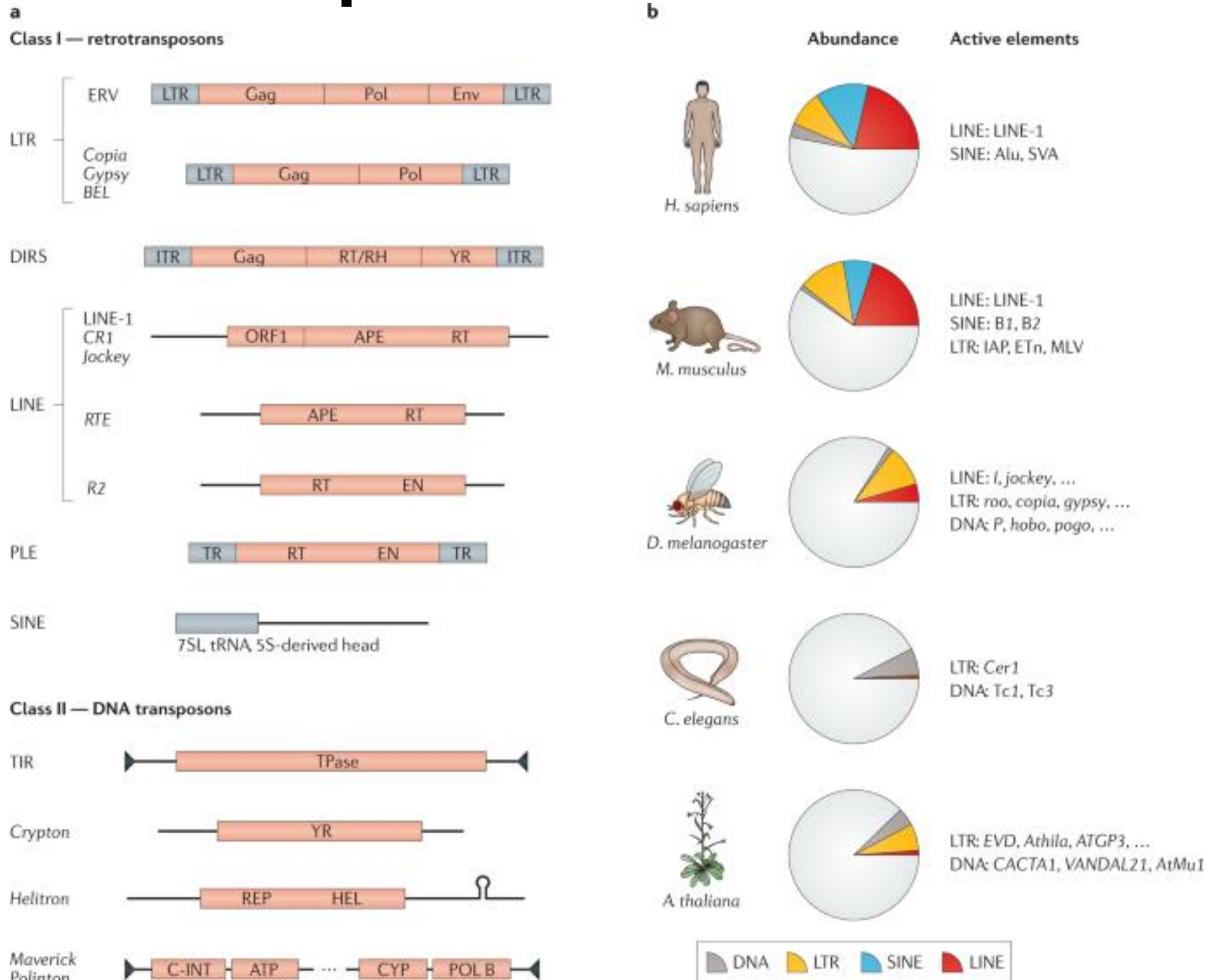
- Many repeats contain ORFs, can be mistaken as genes.
- Repeats should be “masked” before you try to annotate genes
 - ❖ “NNNNN” = hardmasked
 - ❖ “atcg” = softmasked → converted to lower case
 - ❖ Better for downstream BLAST or genome alignment

Solution: Repeat Masking

repeats



Repeats abundance



Two Methods of Repeat Finding

1. Database method

- **RepeatMasker** (repeatmasker.org)
 - Blast **RepBase** elements to your genome
 - Good for mammals or model organisms
 - **Ascertainment bias – species have unique repeats**
 - Human: 50% masked with “homo sapiens” repeats.
 - Humpack whale: 38% masked with “mammalia” repeats.
 - Glass lizard: 13% masked with “vertebrate” repeats
 - Platyhelminthes: >30% repeats (**not enough known**)

Two Methods of Repeat Finding

2. De novo method

- RepeatModeler (repeatmasker.org)
 - Blast your genome to itself
 - Models repeats without *a priori* knowledge
 - Good for finding species-specific repeats
 - May miss low-copy number repeats

- General methodology:

1. Run RepeatMasker on the genome using clade-appropriate RepBase library.
2. Run RepeatModeler on the unmasked remainder of the genome.
3. Combine the RepBase library with the *de novo* library and run RepeatMasker again

Now we can proceed to “Gene Annotation”

Methods based on:

Protein homology

- Search against specific DB

- ***Ab-initio* gene prediction software**

- SNAP
- Augustus
 - **Requires training** to your specific species (**HMM**)
- GeneMark
- Glimmer

Expression data

- RNA-Seq
- This is the only direct evidence used

Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

Protein homology

➤ Search against specific DB

Búsqueda utilizando herramientas de alineamiento (Ej. BLAST)

Conservación evolutiva

Similitud con secuencias conocidas

- BLASTn Puntos de referencia genómica
- BLASTx Identificación de homólogos
- No es posible encontrar “nuevos” genes

Uso de información conocida:

Programas de predicción de genes permiten el uso de homología con secuencias conocidas para mejorar las predicciones

Data that can be used:

- Proteins of related species
- ESTs o cDNAs
- Whole genome-to-genome alignment
- RNA-Seq data

Vs

Genome

Protein homology

- Search against specific DB

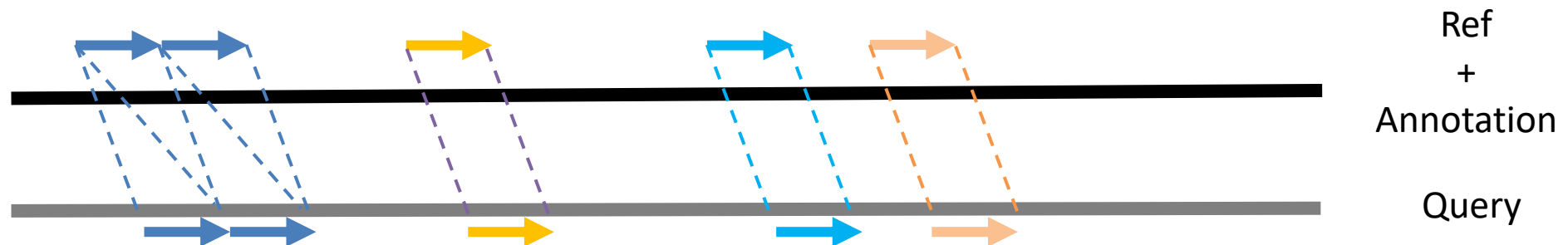
exonerate is a general tool for sequence comparison.

- It uses the **C4** dynamic programming library. It is designed to be both general and fast. It can produce either gapped or ungapped alignments, according to a variety of different alignment models.

GeneWise and Genomewise:

- Start with a PFAM domain HMM
- Replace AA emissions with codon emissions

Gene transfer annotation



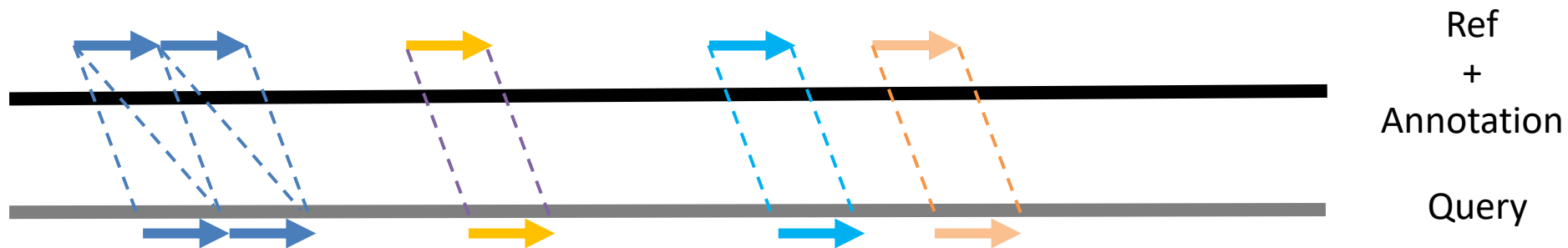
Protein homology

- Search against specific DB

Genome-to-genome alignment:

Requires annotated genome of closely related species

Gene transfer annotation



<https://github.com/TheSEED/RASTtk-Distribution/releases/>




RATT: Rapid Annotation Transfer Tool

<http://ratt.sourceforge.net/>

Chromosomer: a reference-based genome arrangement tool for producing draft chromosome sequences

<https://github.com/gtamazian/chromosomer/wiki/Brief-guide-to-Chromosomer-assembly-process>

Gaik Tamazian , Pavel Dobrynin, Ksenia Krashenninnikova, Aleksey Komissarov, Klaus-Peter Koepfli & Stephen J. O'Brien

• ***Gene Annotation: Ab-initio gene prediction***

Uses likelihoods to find the most likely gene models

method based on **gene content** :
(statistical properties of protein-coding sequence)

- codon usage
- hexamer usage
- GC content
- compositional bias between codon positions
- nucleotide periodicity
- exon/intron size
- ...

and on **signal detection**:

- Promoter
- ORF
- Start codon
- Splice site (Donor and acceptor)
- Stop codon
- Poly(A) tail
- CpG islands...

=> *Ab initio* tools will combine this information through different Probabilistic models: HMM, GHMM, WAM, etc.

These models need to be created if not already existing for your organism => training!

• Gene Annotation: Ab-initio gene prediction

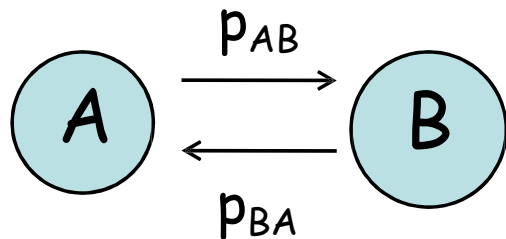
How does the algorithm work?

Hidden Markov Model (HMM)

Markov chain - a linear series of states in which each state is dependent only on the previous state.

HMM - a model that uses a Markov chain to infer the most likely states in data with unknown states ("hidden" states).

A Markov chain has states and transition probabilities:



	A	B
A	0.98	0.02
B	0.4	0.6

A → B

B → A

(implicitly the probability of staying in state A is $1 - p_{AB}$ and the probability of staying in state B is $1 - p_{BA}$)

- ***Gene Annotation: Ab-initio gene prediction***

Hidden Markov Model (HMM)

- We have a Markov chain with appropriate states and known transition probabilities (e.g. inferred from experimentally known genes).
- We have a DNA sequence with unknown states.
- Find the series of Markov chain states with the maximum likelihood for the DNA sequence. **Emission and transition states**
- Solved with the Viterbi algorithm

- ***Gene Annotation: Ab-initio gene prediction***

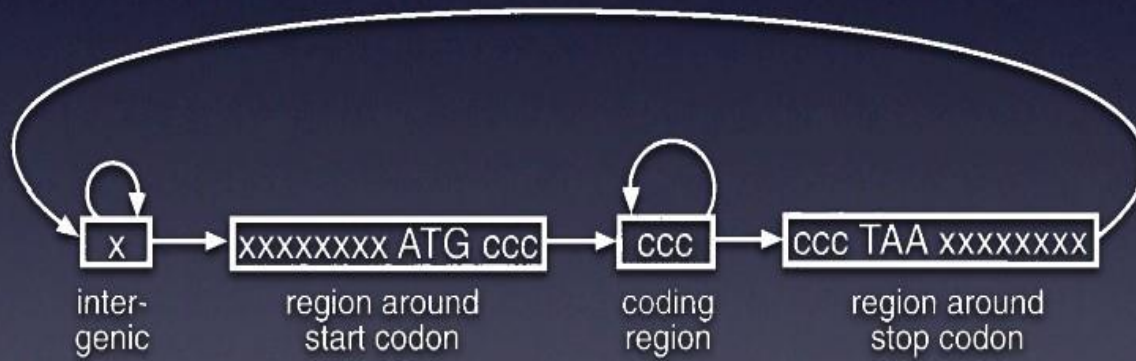
Three classic HMM problems

1. **Evaluation:** given a model and an output sequence, what is the probability that the model generated that output?
2. **Decoding:** given a model and an output sequence, what is the most likely state sequence through the model that generated the output?
3. **Learning:** given a model and a set of observed sequences, how do we set the model's parameters so that it has a high probability of generating those sequences?

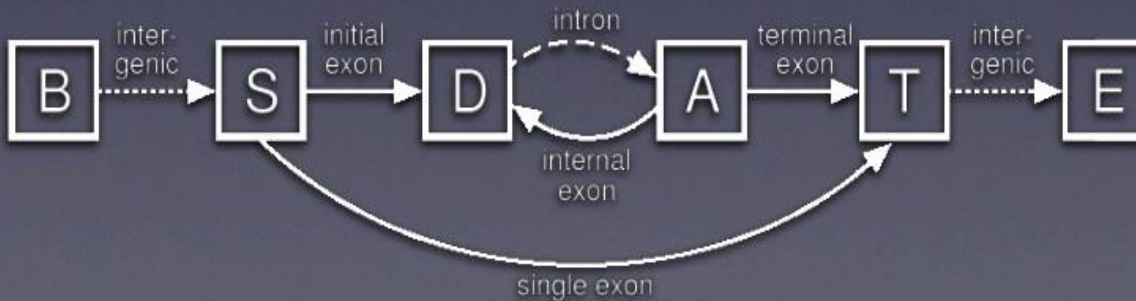
- **Gene Annotation: Ab-initio gene prediction**

How does the algorithm work?

The hidden Markov Models



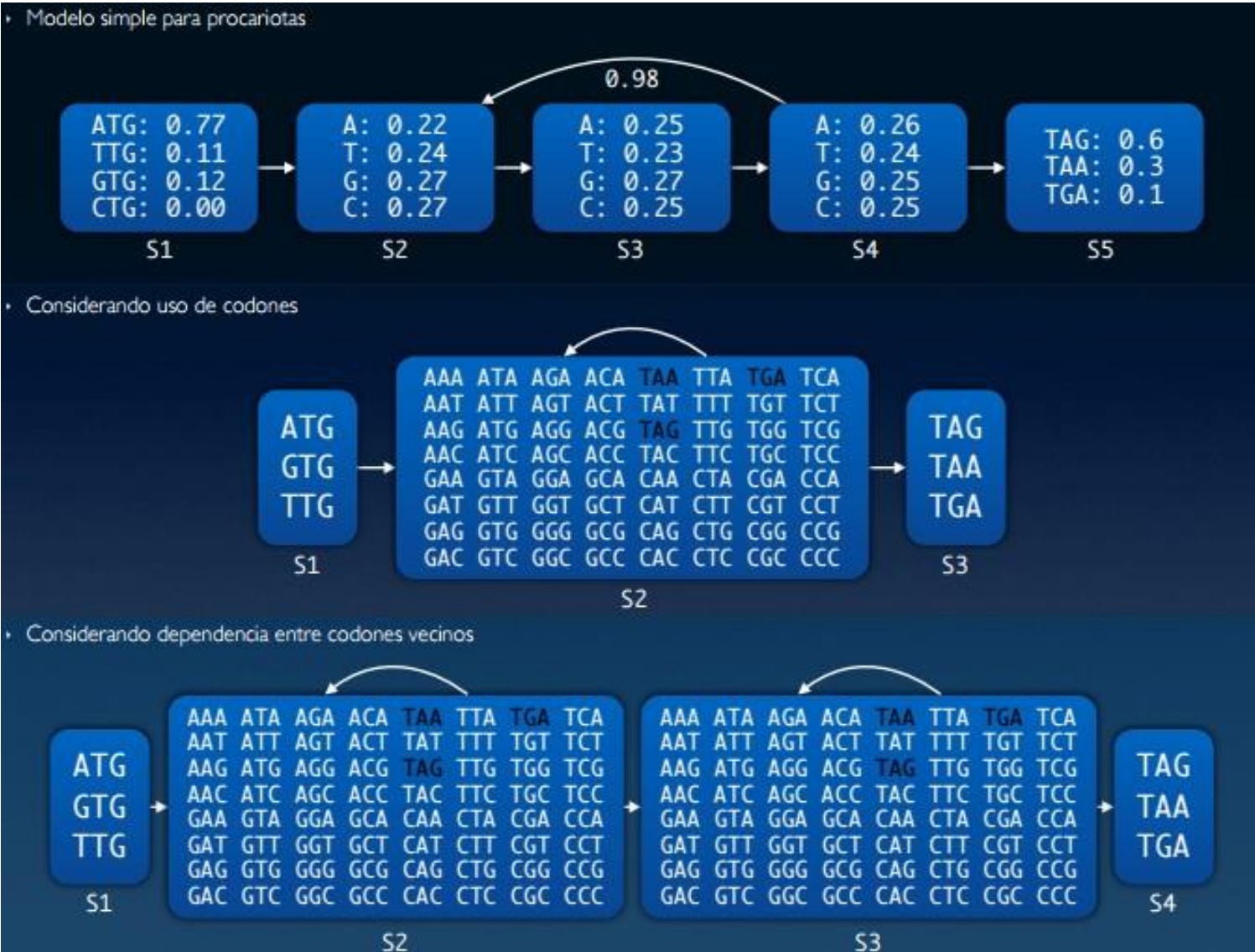
▸ Modelo simple sin Intrones



▸ Modelo simple con intrones exones y señales

• Gene Annotation: Ab-initio gene prediction

How does the algorithm work? The hidden Markov Models

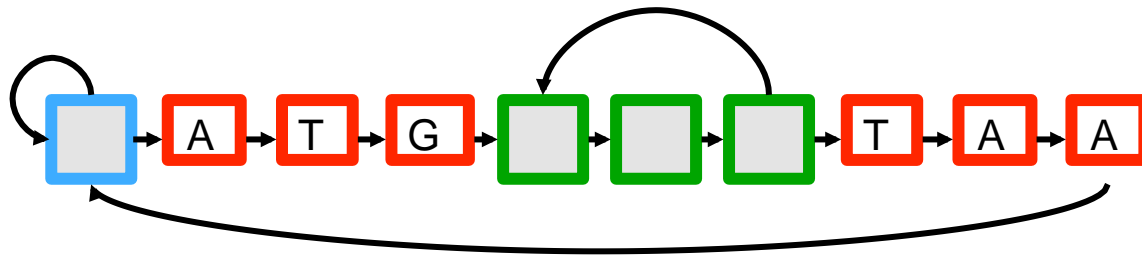


• *Gene Annotation: Ab-initio gene prediction*

HMM for bacterial genomes

- Nucleotides $\{A, C, G, T\}$ are the observables
- Different states generate nucleotides at different frequencies A

simple HMM for unspliced genes:



AAAGCATG CATTTAACGAGA GCA CAA GGG CTC TAATGCCG

- The sequence of states is an annotation of the generated string – each nucleotide is generated in intergenic, start/stop, coding state

- ***Gene Annotation: Ab-initio gene prediction***

Glimmer: HMM for bacterial genomes

- A (very) brief overview of microbial gene-finding
- Evolution of Glimmer
 - Glimmer 1
 - Interpolated Markov Model (IMM)
 - Glimmer 2
 - Interpolated Context Model (ICM)
 - Glimmer 3
 - Reducing false positives with a DP alg for selection
 - Improving coding initiation site predictions

- ***Gene Annotation: Ab-initio gene prediction***

Glimmer: HMM for bacterial genomes

1. Create models that have a probability of generating any given sequence.
2. Evaluate gene/non-genome models against a sequence
3. Train the models using examples of the types of sequences to generate.
4. Use RNA sequencing, homology, or “obvious” genes
5. The "score" of an orf is the probability of the model generating it.
6. Can also use a negative model (i.e., a model of non- orfs) and make the score be the ratio of the probabilities (i.e., the odds) of the two models.
7. Use logs to avoid underflow

- ***Gene Annotation: Ab-initio gene prediction***

Glimmer: HMM for bacterial genomes

- **Glimmer1 & 2 used rules.**
- For **overlapping orfs A and B**, **the overlap region AB** is scored separately:
 1. If AB scores higher in A's reading frame, and A is longer than B, then reject B.
 2. If AB scores higher in B's reading frame, and B is longer than A, then reject A.
 3. Otherwise, output both A and B with a "suspicious" tag.
 4. Also try to move start site to eliminate overlaps.
- Leads to high false-positive rate, especially in high-GC genomes.

- ***Gene Annotation: Ab-initio gene prediction***

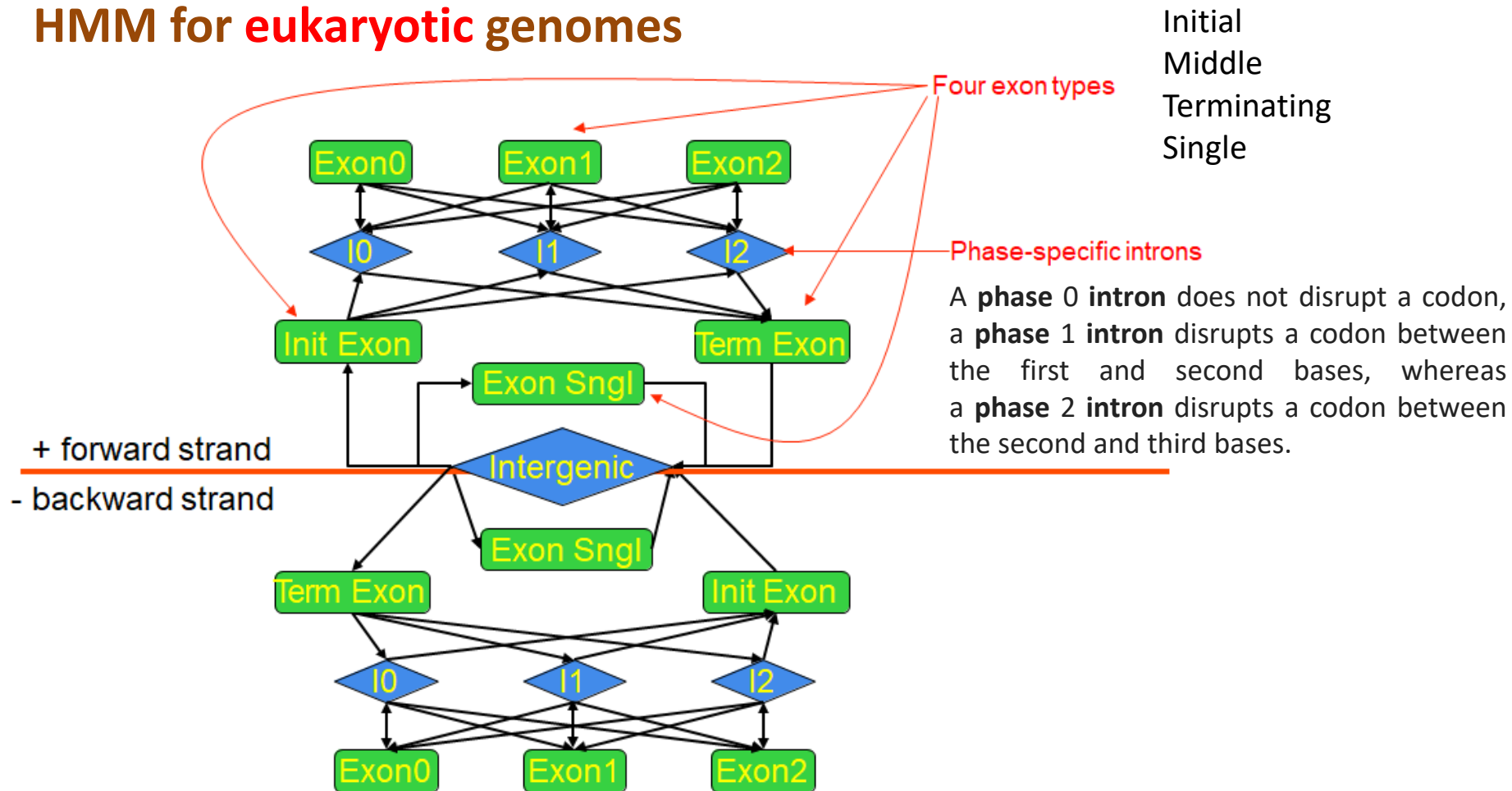
Glimmer: HMM for bacterial genomes

Glimmer3

- Uses a **dynamic programming algorithm** to **choose the highest-scoring set of orfs and start sites**.
 - Similar to the longest increasing subsequence problem of Dynamic Programming
- Not quite an HMM
 - **Allows small overlaps** between genes
 - "small" is user-defined
 - Scores of genes are not necessarily probabilities.
 - Score includes component for likelihood of start site

• *Gene Annotation: Ab-initio gene prediction*

HMM for **eukaryotic** genomes



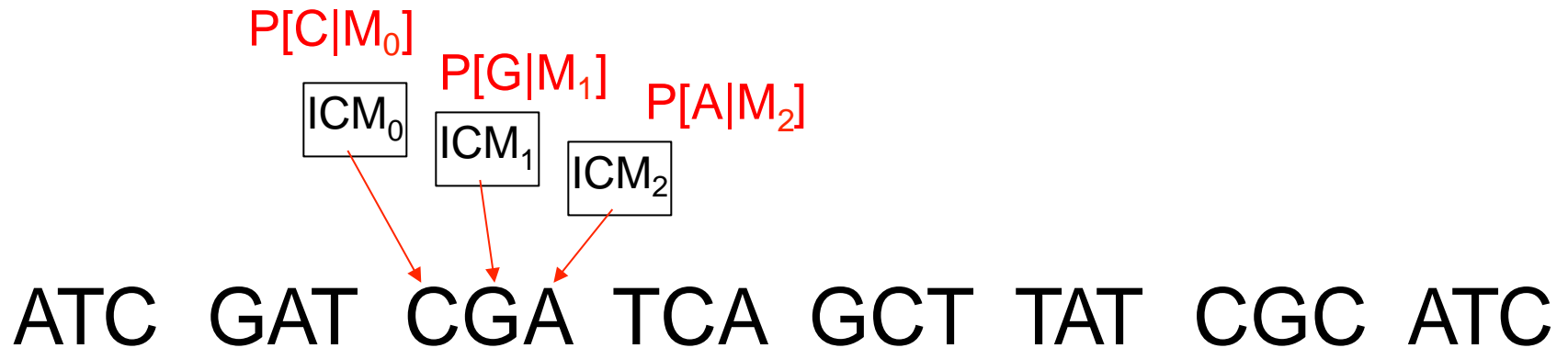
- Uses HMM to model gene structure (explicit length modeling)
- Various models for scoring individual signals
- Can emit a graph of high-scoring ORFS

• *Gene Annotation: Ab-initio gene prediction*

HMM for eukaryotic genomes

Coding vs Non-coding

A three-periodic states uses three states in succession to evaluate the different codon positions, which have different statistics:



The three states correspond to the three phases.

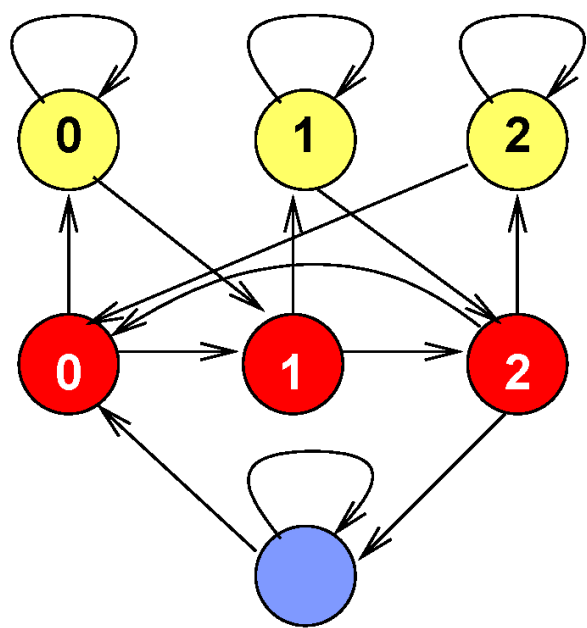
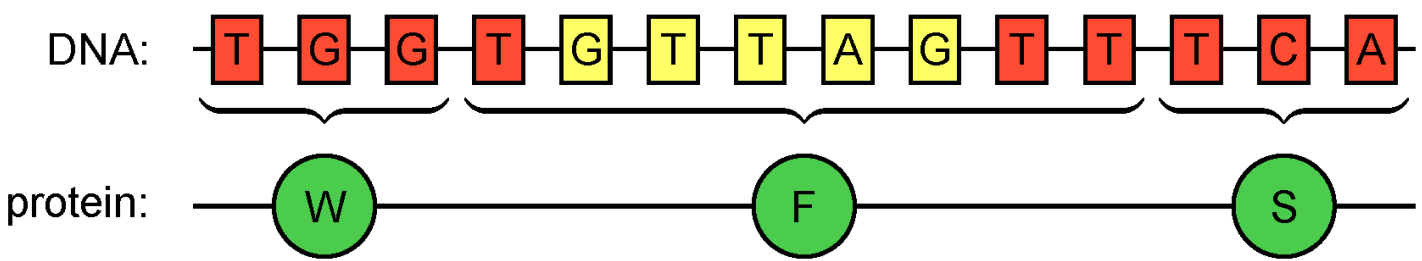
Every base is evaluated in every phase, and the score for a given stretch of (putative) coding DNA is obtained by multiplying the phase-specific probabilities in a mod 3 fashion:

- **Gene Annotation: Ab-initio gene prediction**

HMM for eukaryotic genomes

More complex HMMs: change state transition diagrams

Keep consistent triples across introns:



Necessary to discriminate introns from exons and make the decision of staying in the intron state, exon state

- **Gene Annotation: Ab-initio gene prediction**

HMM for eukaryotic genomes

Identifying Signals In DNA

We slide a fixed-length model or “window” along the DNA and evaluate `score(signal)` at each point:

Signals – short sequence patterns in the genomic DNA that are recognized by the cellular machinery.

Signal sensor

...ACTGATGCGCGATTAGAGTCATGGCGATGCATCTAGCTAGCTATATCGCGTAGCTAGCTAGCTGATCTACTATCGAGC...

When the `score` is greater than some threshold (determined empirically to result in a desired sensitivity), we remember this position as being the potential site of a signal.

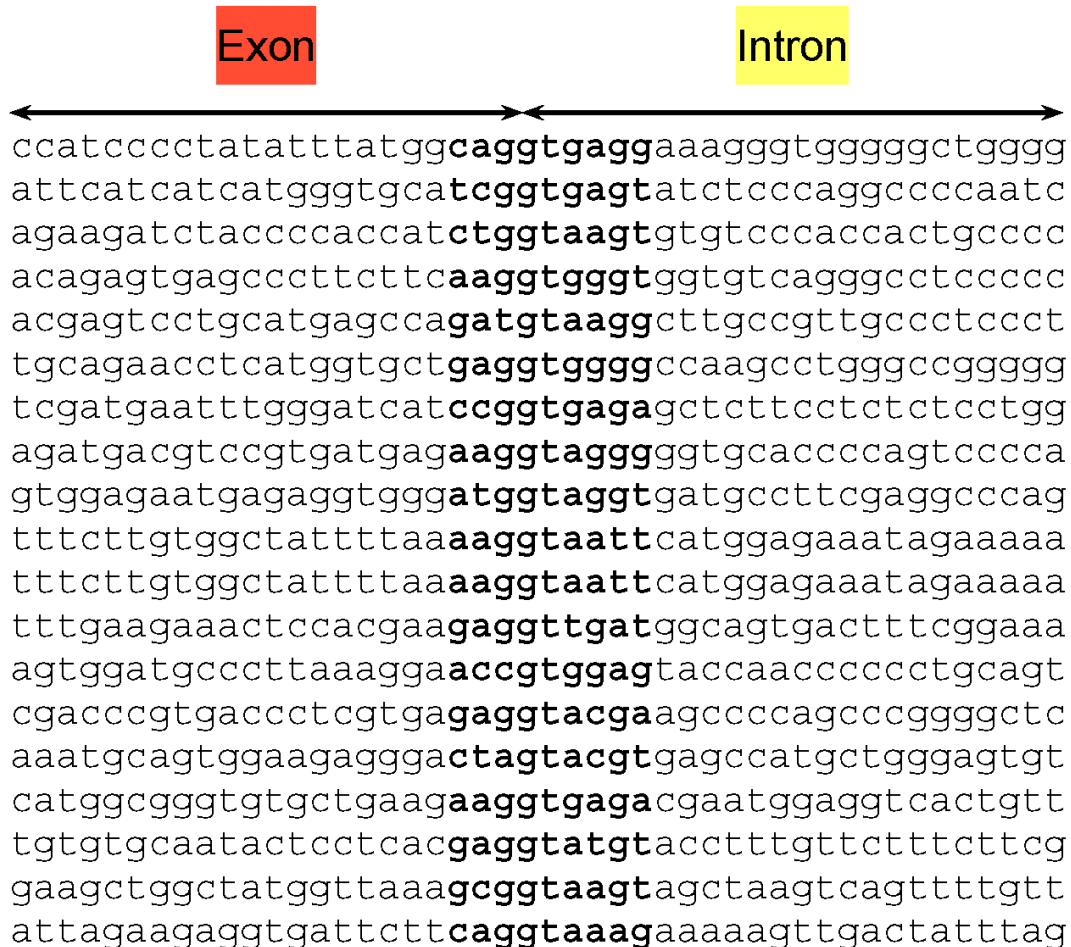
The most common signal sensor is the Weight Matrix:

A = 31% T = 28% C = 21% G = 20%	A = 18% T = 32% C = 24% G = 26%	A 100%	T 100%	G 100%	A = 19% T = 20% C = 29% G = 32%	A = 24% T = 18% C = 26% G = 32%
--	--	------------------	------------------	------------------	--	--

• *Gene Annotation: Ab-initio gene prediction*

HMM for eukaryotic genomes

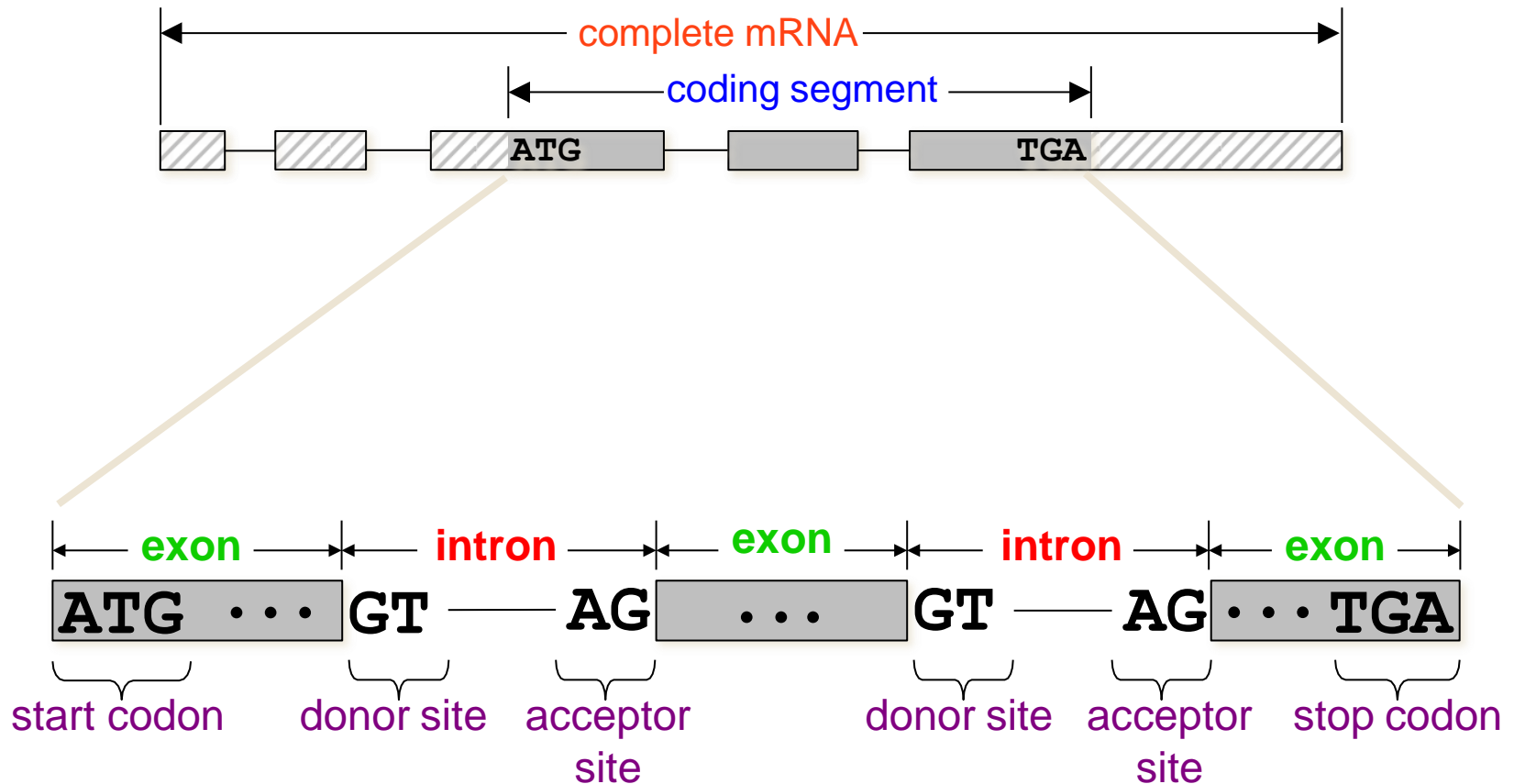
More complex HMMs: remember those signals?



- **Gene Annotation: Ab-initio gene prediction**

HMM for eukaryotic genomes

Eukaryotic Gene Syntax



Regions of the gene outside of the CDS are called **UTR's** (*untranslated regions*), and are mostly ignored by gene finders, though they are important for regulatory functions.

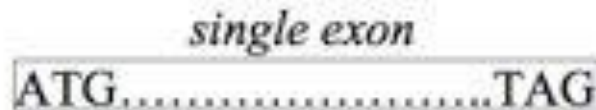
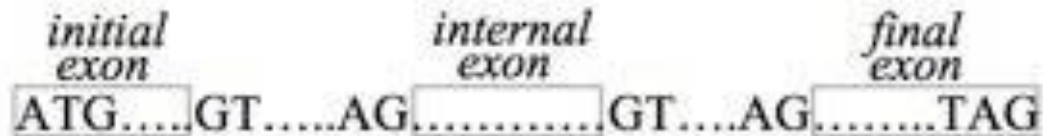
• Gene Annotation: Ab-initio gene prediction

HMM for eukaryotic genomes

Types of Exons

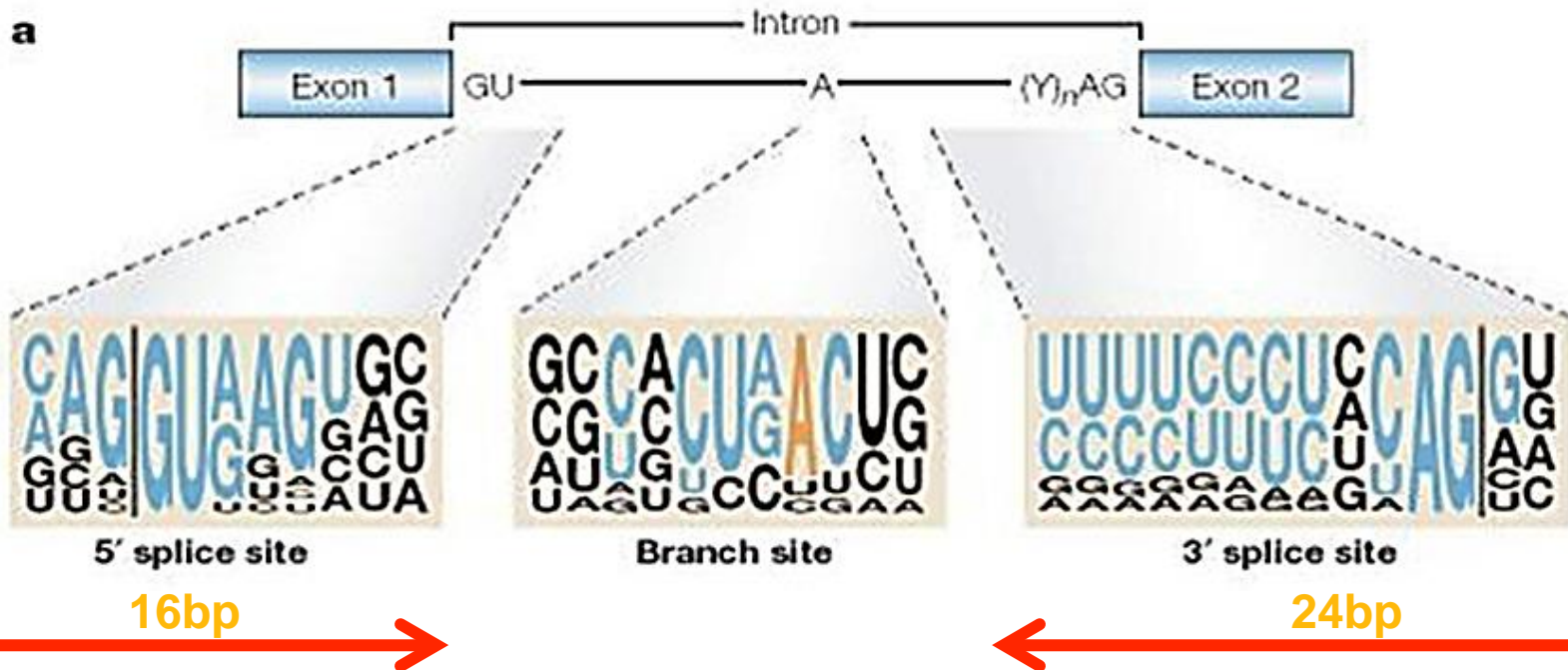
Three types of exons are defined, for convenience:

- *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:



• Gene Annotation: Ab-initio gene prediction

HMM for eukaryotic genomes **Splice site prediction**



The splice site score is a combination of:

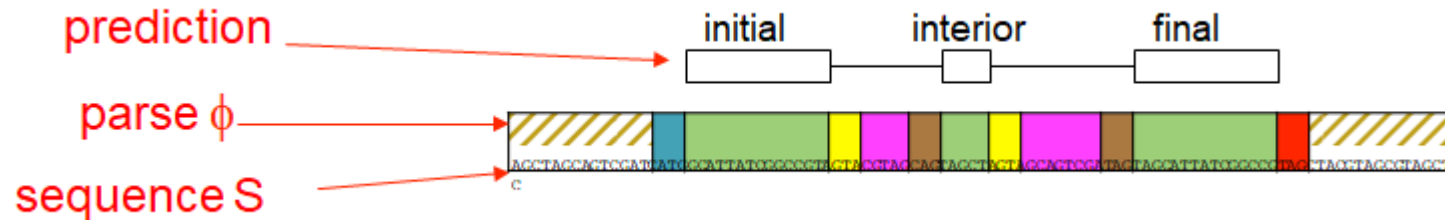
1. first or second order in homogeneous Markov models on windows around the acceptor and donor sites
2. Maximal dependence decomposition (MDD) decision trees
3. longer Markov models to capture difference between coding and non-coding on opposite sides of site (optional)
4. maximal splice site score within 60 bp (optional)

- **Gene Annotation: Ab-initio gene prediction**

HMM for eukaryotic genomes

Gene Prediction with a HMM

Given a sequence S , we would like to determine the parse ϕ of that sequence which segments the DNA into the most likely exon/intron structure:

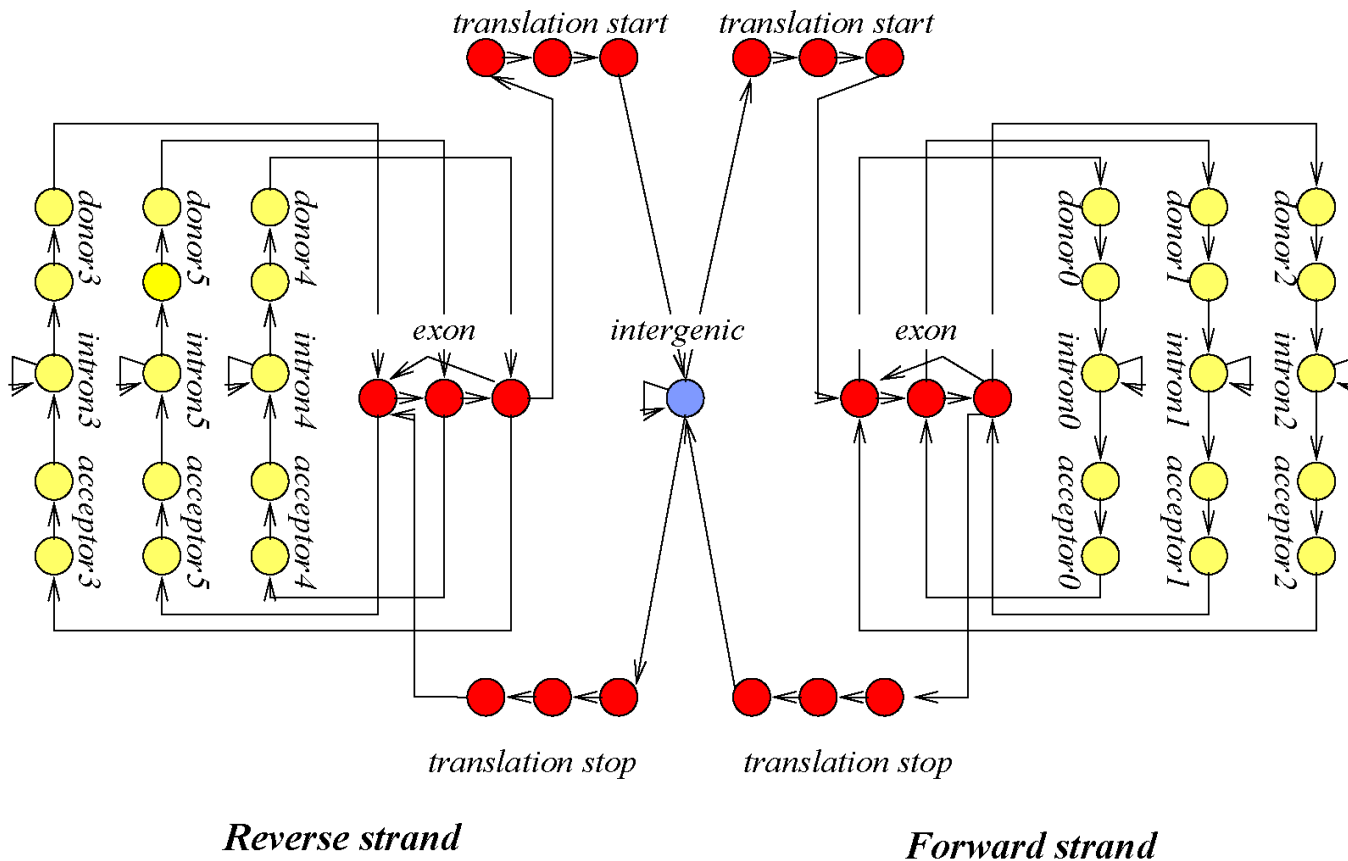


1. The parse ϕ consists of the coordinates of the predicted exons,
2. Identify precise sequence of states during the operation
3. emits an entire feature such as an exon or intron.

• *Gene Annotation: Ab-initio gene prediction*

HMM for eukaryotic genomes

Example of an HMM for gene finding



• *Gene Annotation: Ab-initio gene prediction*

HMM for eukaryotic genomes

Summary

- HMMs are useful tool for gene finding
- Viterbi algorithm used for inference
- Higher order states model dependencies among adjacent letters
- Complex state transition diagrams incorporate biological knowledge

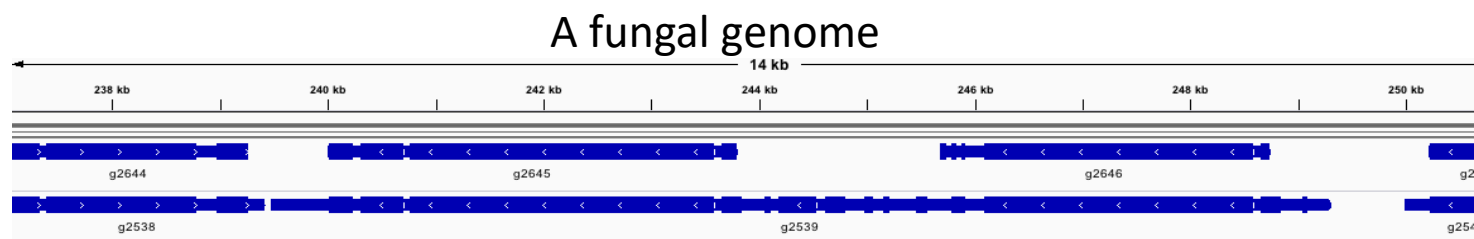
This is just a beginning

- More complex models used in practice: lengths of exons/introns, complex signal models, experimental information, multiple species, . . .
- HMMs for other bioinformatics tasks: protein secondary structure, protein families, segmenting genome, . . .

• *Gene Annotation: Ab-initio gene prediction*

Training *ab-initio* gene-finders

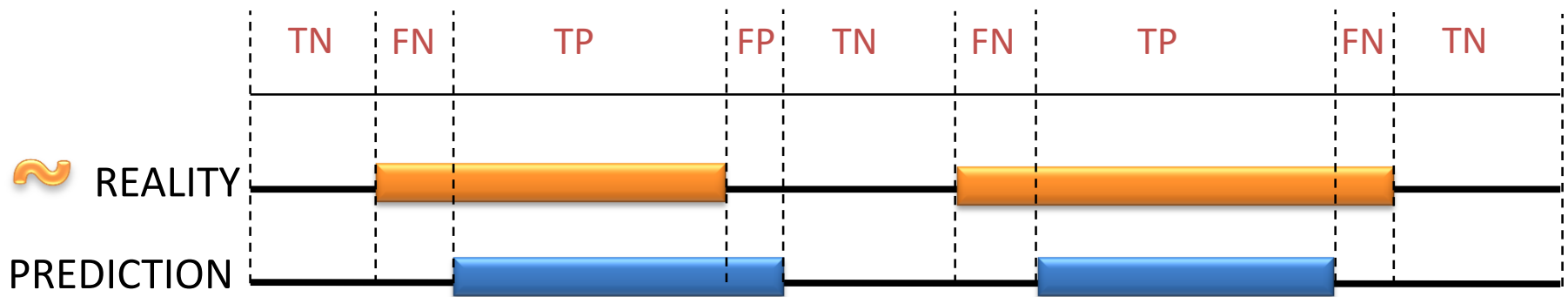
- Some gene-finders **train themselves**, others need a separate training procedure
- Around **1000 already known genes** are usually needed to train the gene-finder
- These "known" genes can be **inferred** from aligned transcripts or proteins
- **The quality of the gene-finder results hugely relies on the quality of the training!**



• *Gene Annotation: Ab-initio gene prediction*

Assessing quality

Assess the quality of an annotation:



Sensitivity is the proportion of true predictions compared to the total number of correct genes (including missed predictions)

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

Specificity is the proportion of true predictions among all predicted genes (including incorrectly predicted ones)

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

Ab Initio methods can approach 100% sensitivity, however as the sensitivity increases, accuracy suffers as a result of increased false positives.

• *Gene Annotation: Ab-initio gene prediction*

Popular tools:

- **SNAP** Works ok, easy to train, not as good as others especially on longer intron genomes.
- **Augustus** Works great, hard to train (but getting better).
- **GeneMark-ES** Self training, no hints, buggy, not good for fragmented genomes or long introns (Best suited for Fungi).
- **FGENESH** Works great, costs money even for training.
- **GlimmerHMM** (Eukaryote)
- **GenScan**
- **Gnomon** (NCBI)



Supported
by MAKER

• *Gene Annotation: Ab-initio gene prediction*

Strengths :

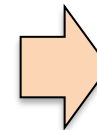
- Fast and easy means to identify genes
- Annotate unknown genes
- “Exhaustive” annotation
- Need no external evidence

Limits :

- No UTR*
- No alternatively spliced transcripts*
- Over prediction (exons or genes)
- **Training** needed to perform well

Unless it has been trained
with RNA-seq / hints data

- Split single gene into multiple predictions
- Fused with neighboring genes
- Less accurate than homology based method:
 - Exon boundaries
 - Splicing sites

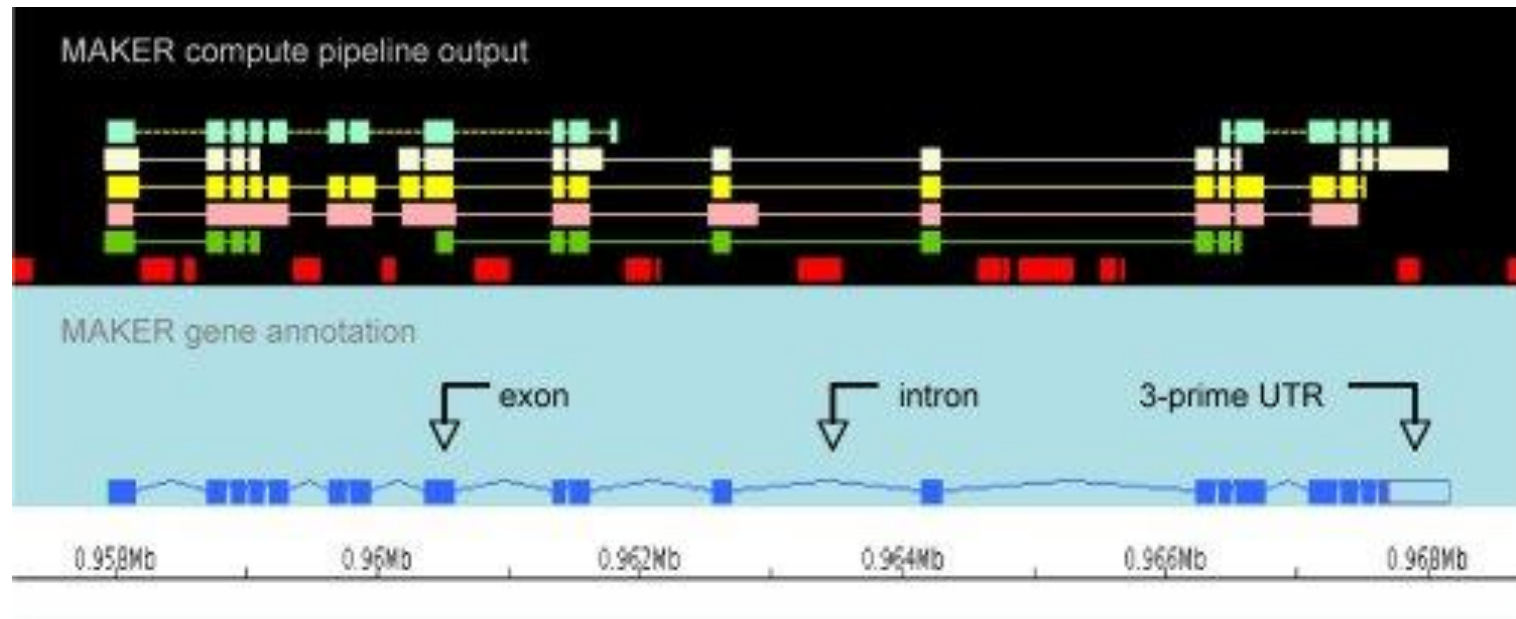


Hybrid
method

- **Gene Annotation: Ab-initio gene prediction**

Pipeline

MAKER (Holt and Yandell, 2011)



SNAP *ab-initio* Gene Prediction
EST Alignment - EXONERATE
Protein Alignment - EXONERATE
Protein Alignment - BLASTX

EST Alignment - BLASTN
Repeats
MAKER gene annotation

- ***Gene Annotation: Ab-initio gene prediction***

Pipeline



Prokka

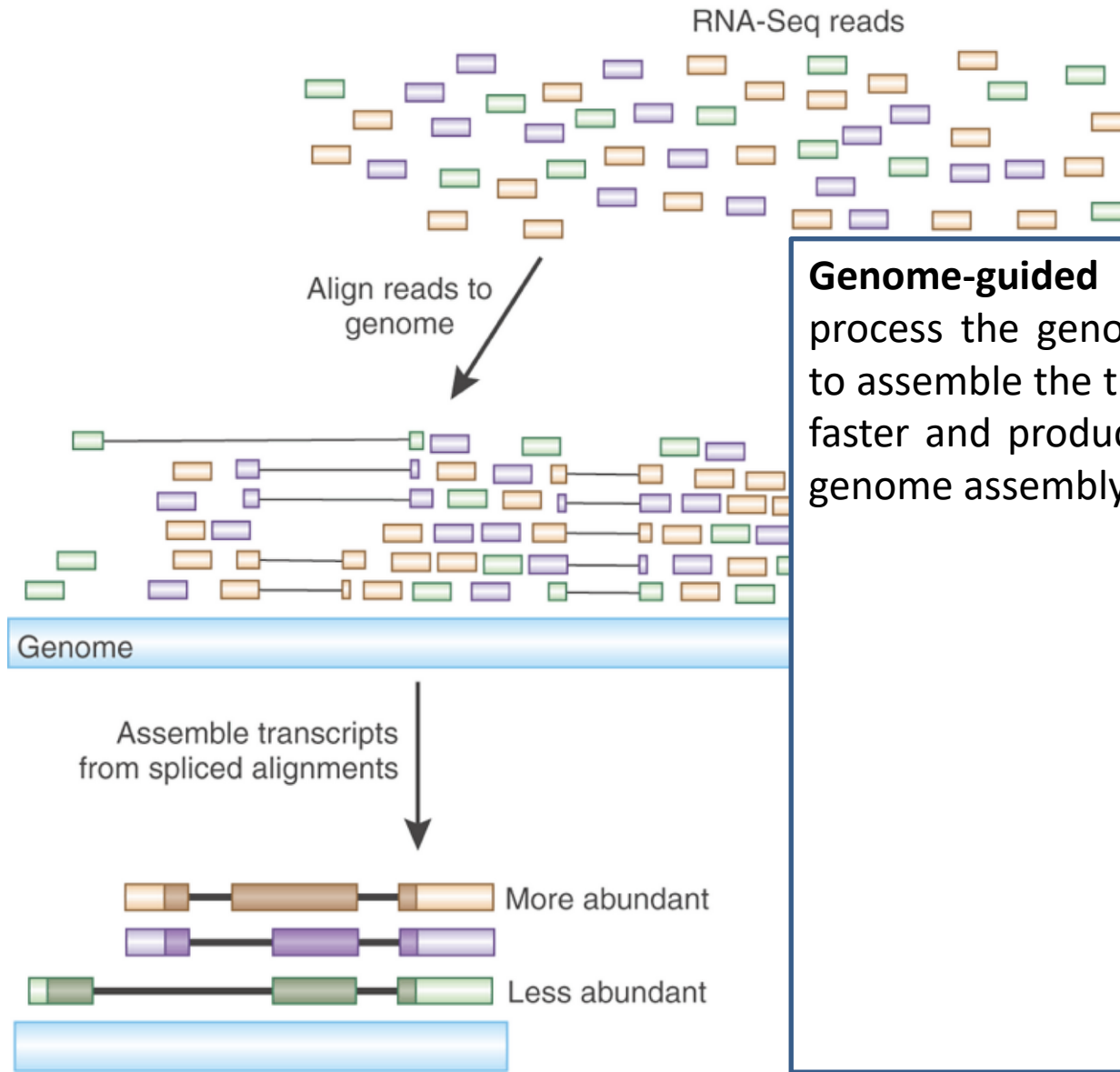
Description

Prokka is a software tool for the rapid annotation of prokaryotic genomes. A typical 4 Mbp genome can be fully annotated in less than 10 minutes on a quad-core computer, and scales well to 32 core SMP systems. It produces GFF3, GBK and SQN files that are ready for editing in Sequin and ultimately submitted to Genbank/DDJB/ENA.



Expression data

- RNA-Seq
- This is the only **direct evidence used**



Genome-guided transcriptome assembly: process the genomic alignments (BAM files) to assemble the transcripts. It is usually much faster and produces better results when the genome assembly is of good quality.

BOWTIE

Blat

HiSat2

STAR

TopHat2

StringTie

Whippet

CuffLinks

Expression data

- RNA-Seq
- This is the only direct evidence used



De novo transcriptome assembly: take input the FASTQ files and it loads them into the RAM memory to a fast exploration of the possible solutions to assemble the reads as transcripts, therefore it requires large amounts of RAM memory. It is particularly useful when no good quality genome assembly is available.

Trinity

RNASpades

SOAPdenovo-Trans

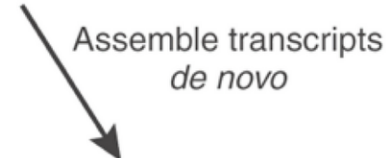
Velvet/Oases

Trans-ABYSS

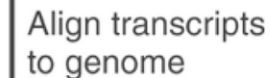
BinPacker

SeqMan NGen

Assemble transcripts
de novo



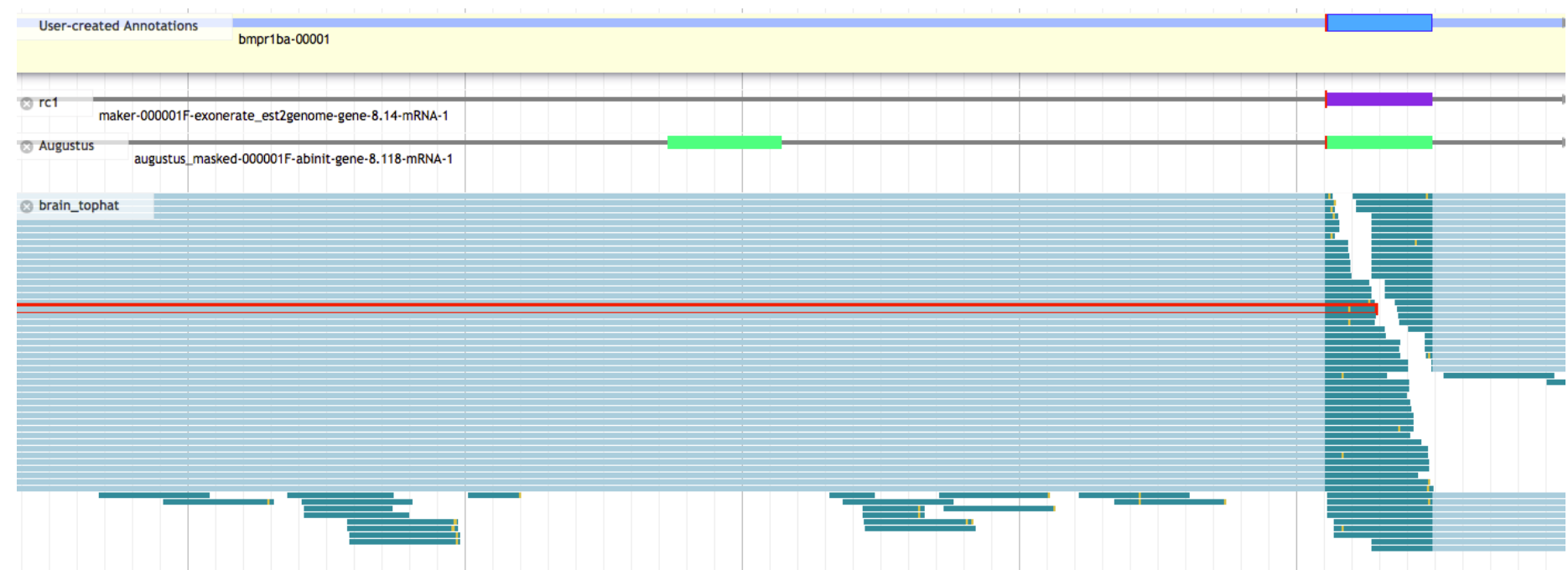
Align transcripts
to genome



Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

Hybrid (*evidence-drivable gene predictors*) approaches incorporate hints in the form of EST alignments or protein profiles to increase the accuracy of the gene prediction.



GenomeScan : The blast hits have to be converted into a probability, but this is made relatively easily, as the E-values give guidance... (Understanding Bioinformatics, Par Marketa J. Zvelebil, Jeremy O. Baum). It's a version based on genscan.

EuGene* can be seen as a combiner because collect information about splice sites and ATG has to be done outside the program.

Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

Hybrid (*evidence-drivable gene predictors*) approaches incorporate hints in the form of EST alignments or protein profiles to increase the accuracy of the gene prediction.

GenomeScan	Blast hit used as extra guide
Augustus	16 types of hints accepted (gff): start, stop, tss, tts, ass, dss, exonpart, exon, intronpart, intron, CDSpert, CDS, UTRpart, UTR, irpart, nonexonpart.
GeneMark-ET	EST-based evidence hints
GeneMark-EP	Protein-based evidence hints
SNAP	Accepts EST and protein-based evidence hints.
Gnomon	Uses EST and protein alignments to guide gene prediction and add UTRs
FGENESH+	Best suited for plant
EuGene*	Any kind of evidence hints. Hard to configure (best suited for plant)

GenomeScan : The blast hits have to be converted into a probability, but this is made relatively easily, as the E-values give guidance... (Understanding Bioinformatics, Par Marketa J. Zvelebil,Jeremy O. Baum). It's a version based on genscan.

EuGene* can be seen as a combiner because collect information about splice sites and ATG has to be done outside the program.

Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

Hybrid (*evidence-drivable gene predictors*) approaches incorporate hints in the form of EST alignments or protein profiles to increase the accuracy of the gene prediction.

The BRAKER1 gene finding pipeline:

BRAKER1: Unsupervised RNA-Seq-Based Genome Annotation with GeneMark-ET and AUGUSTUS

Katharina J. Hoff et *al.*

Bioinformatics (2016) 32 (5): 767-769. doi: 10.1093/bioinformatics/btv661

- BRAKER1 was more accurate than MAKER2 when it is using RNA-Seq as sole source for training and prediction.
- BRAKER1 does not require pre-trained parameters or a separate expert-prepared training step.

Hybrid approaches and pipelines:

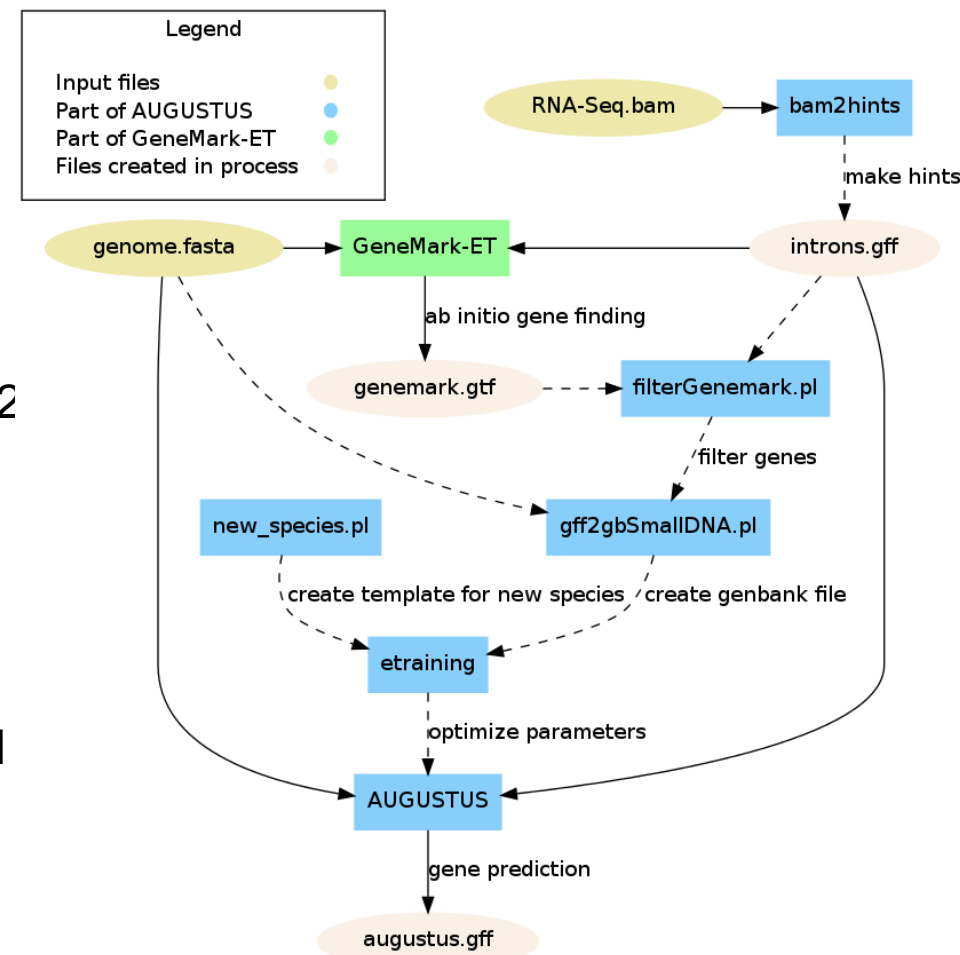
- Ab initio tools with the ability to integrate external evidence/hints

BRAKER1: Unsupervised RNA-Seq-Based Genome Annotation with GeneMark-ET and AUGUSTUS

Katharina J. Hoff et al. Bioinformatics (2016) 32 (5): 767-769. doi: 10.1093/bioinformatics/btv661

The BRAKER1 gene finding pipeline:

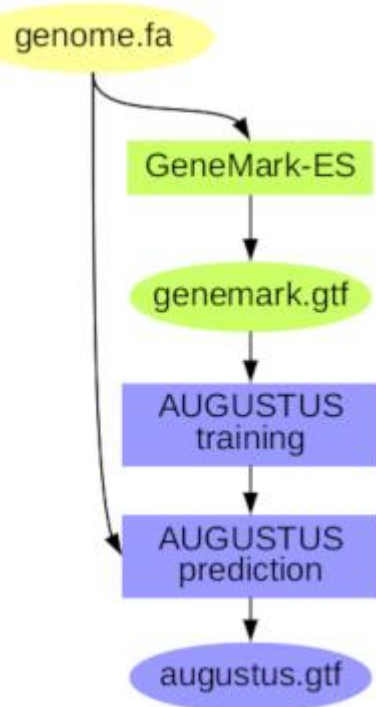
- BRAKER1 is more accurate than MAKER2 when it is using RNA-Seq as sole source for training and prediction.
- BRAKER1 does not require pre-trained parameters or a separate expert-prepared training step.



Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

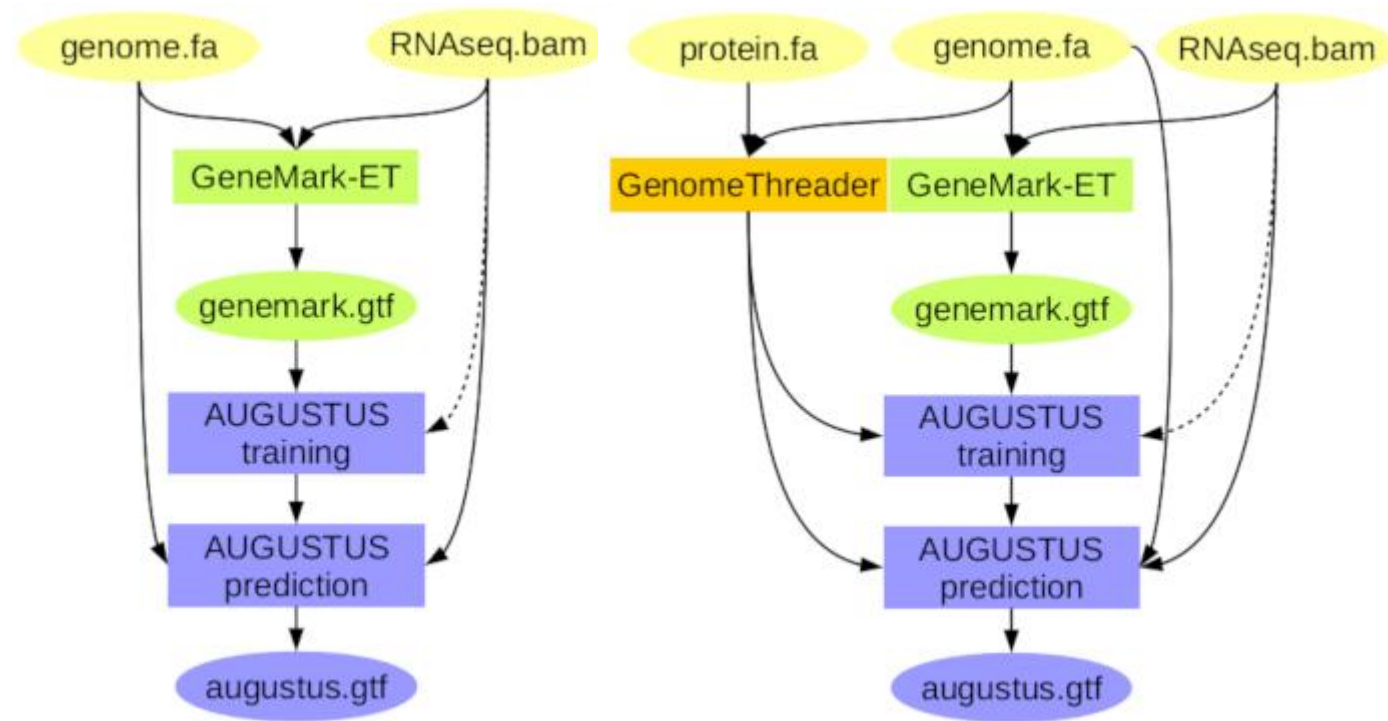
Ab initio



GeneMark-ES is trained on the genome sequence alone.

Long genes predicted by GeneMark-ES are selected for training AUGUSTUS. Final predictions by AUGUSTUS are *ab initio*.

Hybrid



Training GeneMark-ET supported by RNA-Seq spliced alignment information, or/and GenomeThreader supported by protein alignments then prediction with AUGUSTUS with that same spliced alignment information.

Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

MAKER2

MAKER – developed as an easy-to-use alternative to other pipelines

- can be used pure evidence-based, pure *ab initio*, or evidence-driven (on the fly) *ab initio*.
- add UTR when ESTs are supplied.
- Evidence based chooser : select post processed gene model which is most consistent with evidence (protein / EST / RNAseq)

Advantages over competing solutions:

- Easy to use and to configure
- Almost unlimited **parallelism** built-in (limited by data and hardware)
- Largely independent from the underlying system it is run on
- Everything is run through one command, no manual combining of data/outputs
- Follows common standards, produces GMOD compliant output
- **Annotation Edit Distance (AED) metric for improved quality control**
- Provides a mechanism to train and retrain *ab-initio* gene predictors
- Annotations can be updated by re-launching Maker with new evidence

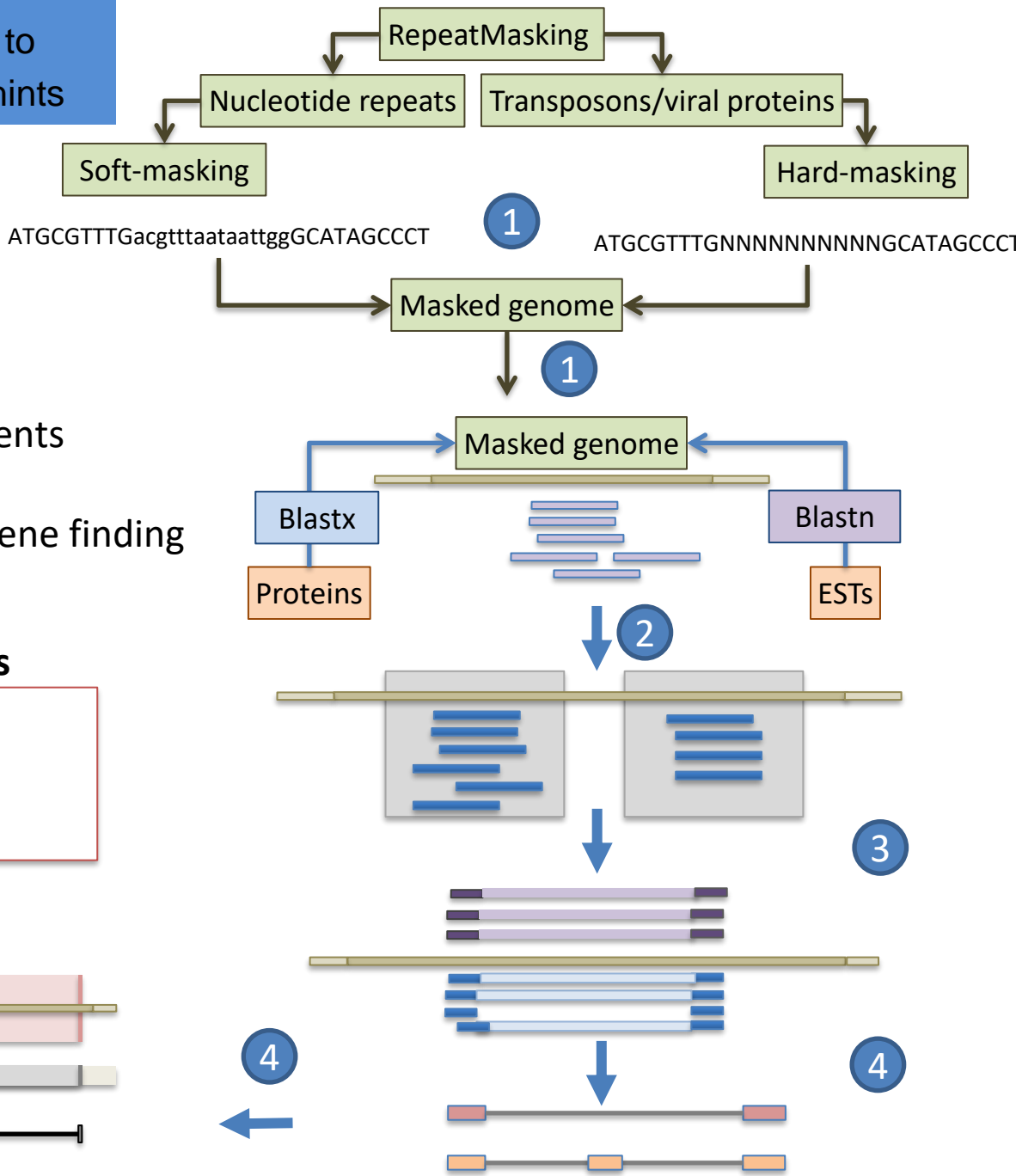
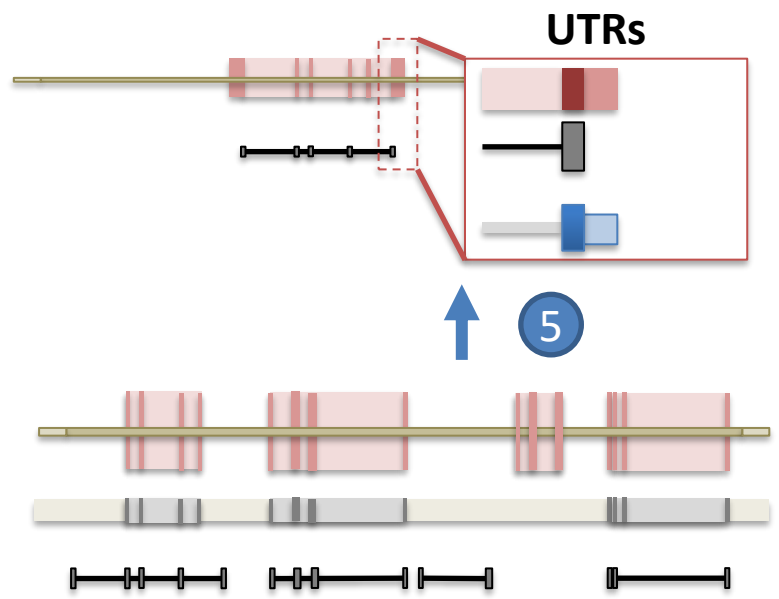
But how does Maker work exactly?

Hybrid approaches and pipelines:

- Ab initio tools with the ability to integrate external evidence/hints

MAKER2 steps

1. Step 1: Raw compute phase
2. Step 2: Filter and cluster alignments
3. Step 3: Polishing alignments
4. Step 4: Synthesis and *ab-initio* gene finding
5. Step 5: Annotate



How to know how good are my annotations?

Tool	Consensus based chooser	Evidence based chooser	weight of different sources	Comment
A) select the prediction whose structure best represents the consensus				
JIGSAW	X			
B) choose the best possible set of exons and combine them in a gene model				
EVM Evidence modeller	X	X	X	User can set the expected evidence error rate manually or/and learn from a training set
Evigan	X		X	Unsupervised learning method
Ipred		X		Does not require any a priori knowledge Can also combine only evidences to create a gene model

Strength => They improve on the underlying gene prediction models

How to choose Method:

- Scientific question behind (need of a conservative annotation vs exhaustive)
- Species dependent (plant / Fungi / eukaryotes)
- phylogenetic relationship of the investigated genome to other annotated genomes (Terra incognita, close, already annotated).
- Data available (hmm profile, RNAseq, etc...)
- Depending on computing resources (*ab initio* ~ hours < vs > pipeline ~ weeks)
- effort versus accuracy

Other genome features

Feature type	DB associated	Tool example	approach
ncRNA	Rfam	infernai	HMM + CM
tRNA	Sprinzl database	tRNAscan-SE	CM + WMA
snoRNA		snoscan	HMM + SCFG
miRNA	miRBase	Splign	sequence alignment
		miR-PREFeR (for plant)	Based on expression patterns
Repeats	Repbase, Dfam	repeatMasker	HMM, blast
Pseudogenes		pseudopipe	homology-based (blast)
...			

Visualization / Manual curation

Selection of most common visualization or/and Manual curation tools

Name	Standalone	Web tool	Manual curation	year	comment
Artemis	X		X	2000	Can save annotation in EMBL format
IGV	X			2011	Popular
Savant	X			2010	Sequence Annotation, Visualization and ANalysis Tool. enable Plug-ins
Tablet	X		X	2013	
IGB	X			2008	enable Plug-ins. Can load local and remote data (dropbox, UCSC genome, etc)
Jbrowse		X		2010	GMOD (successor of Gbrowse)
Web Apollo		X	X	2013	Active community (gmod). Based on Jbrowse. Real-time collaboration
UCSC		X		2000	A large amount of locally stored data must be uploaded to servers across the internet
Ensembl genome browsers		X		2002	A large amount of locally stored data must be uploaded to servers across the internet