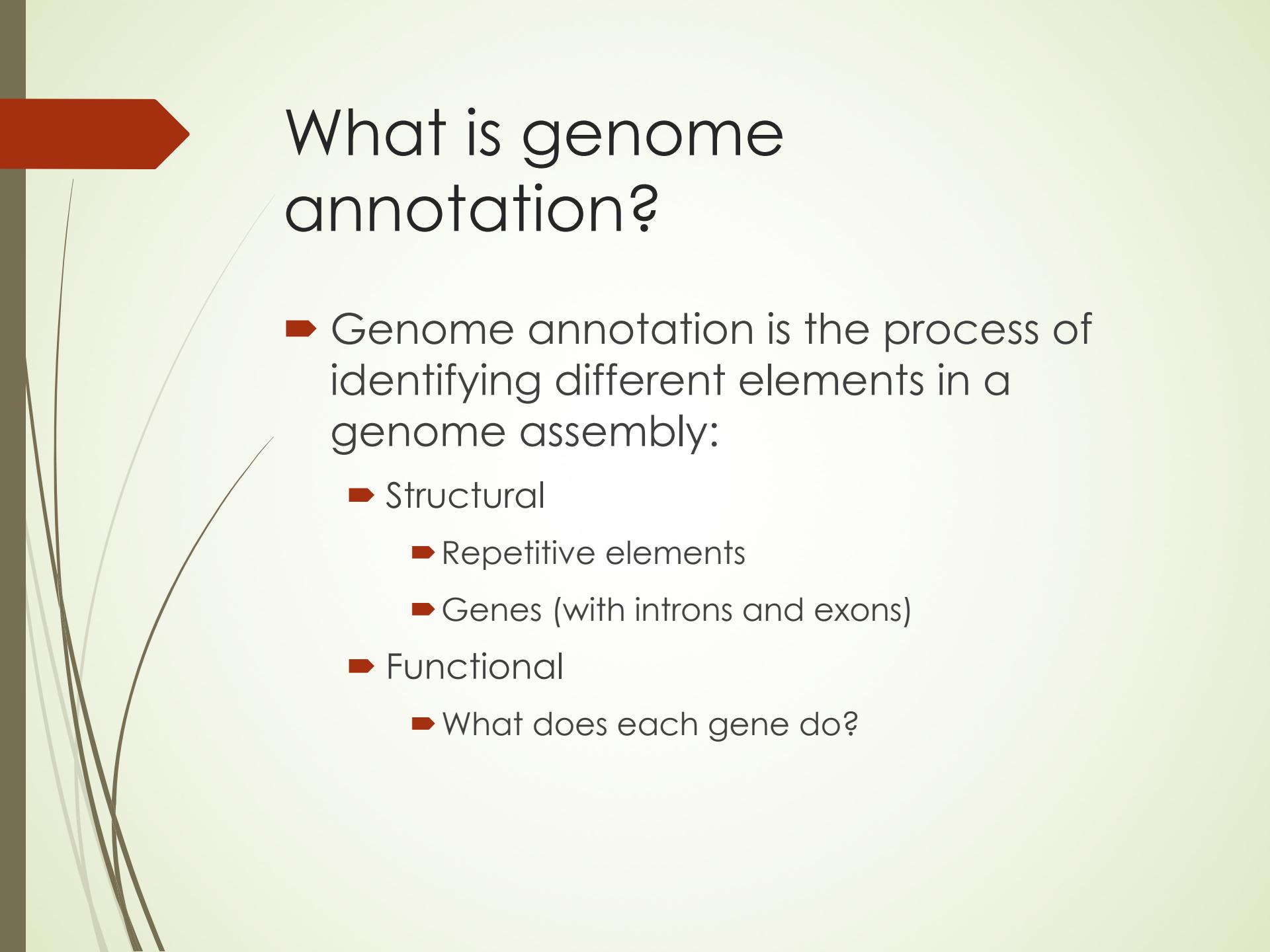


# Genome Annotation

Mirian T. N. Tsuchiya

Data Science Postdoctoral Fellow  
Data Science Lab - OCIO



# What is genome annotation?

- ▶ Genome annotation is the process of identifying different elements in a genome assembly:
  - ▶ Structural
    - ▶ Repetitive elements
    - ▶ Genes (with introns and exons)
  - ▶ Functional
    - ▶ What does each gene do?

# Gene prediction: ab initio vs evidence driven

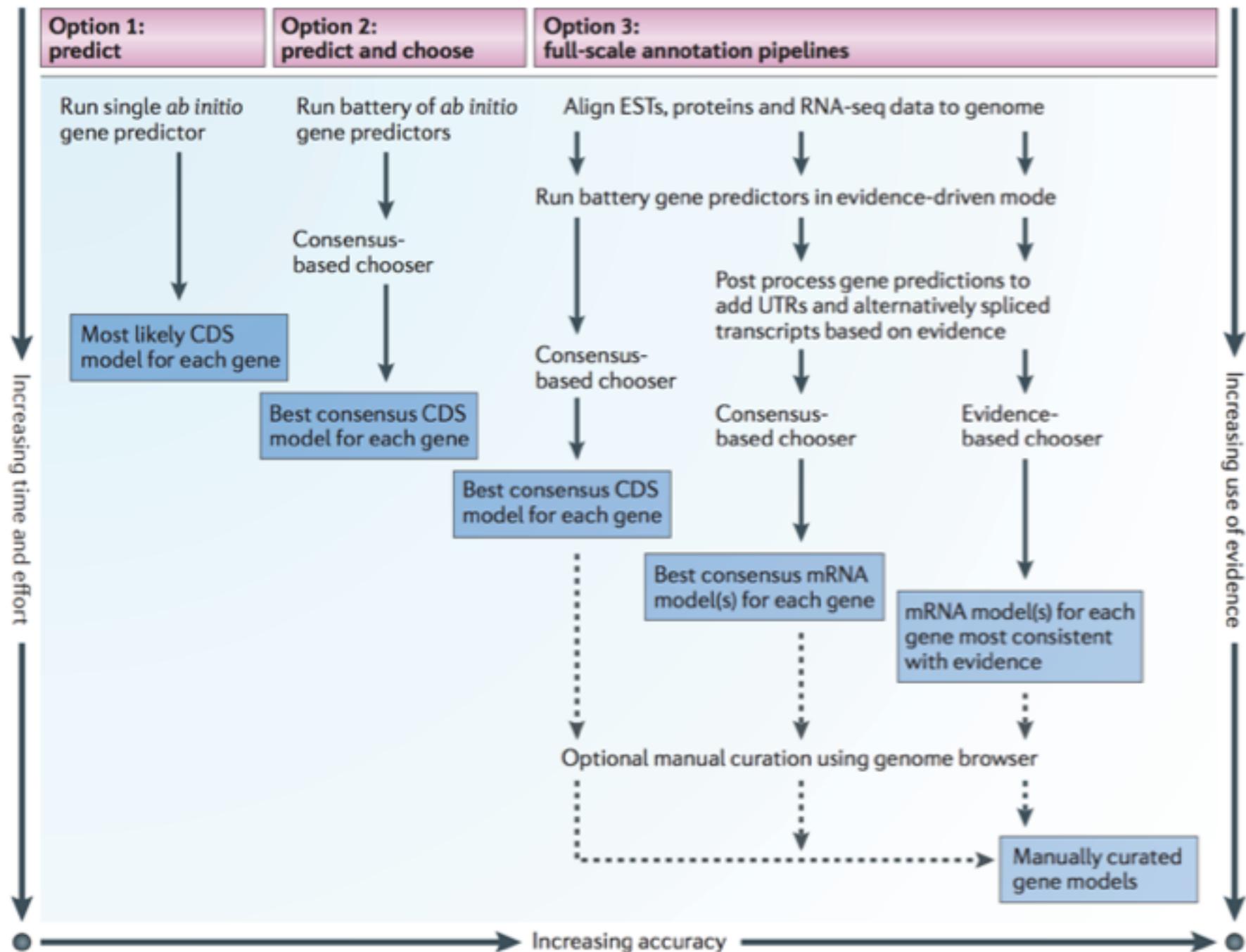
## ab initio

- use only the query sequence

## evidence- driven

- use external  
evidence

**Combining both approaches is the best option**



# Chromosome 3R: 23,049,569-23,054,170



Chr. 3R



ATGTTGCAAAATATGACAACCTTTAATAAAATTCACTTTGGCTTTTGACAGAGTATGTTGTTGGTAGACAAAAGTTACGTAGTCACATAGAACACTGCAGTT  
AACAAACAGTATACTATCAAAAATATAAAATGTTCTTTCGGCCATCGAACATGCCAAAAGGGGAATATCACCGTTAACGCCAACAGAACAAGGACTAATACAA  
ACTTAAGATACTAATAGCTATGTAAGTATACCCCCACAGATGCAACACATTTCAAAGGTATCTGATACTGGGAGATTGTCGACCAGATCAGAAGTCATGGCAGCTCGTA  
GGCACGTTCTGAAAGGAAAGTCGTAAGTAAAGCTTAATCTCTTGTCCCTCGATTCACTGCTTACTCTTAAGGCTCAGCAATTGTTGGCATGGCAGCTAGATTCT  
CTTGGTTGGGATTGCGCAGGACTTCAGCCGCCCTTAATCATGCCCTCGATGCTCTGCAGCAGCCATTCTTATTGGGGCACGTAATAGCCAGACGGTAATGCCA  
TCCGGTGTCCGCCACACGTTCCACTCTGCGGGAGTCGCGACTCTCGGTATCGCTCTCGTCTTATAGCAATGCACCGACCCACCTGGACGGC  
TCATCTCTCGCGGCTTCCGGCACGGACTTGCCTCTGCTATTGAGCTGGCCTTGGATGCGATCCGCTTGGGACACGTACATATTAAACATGTATAAC  
ATGCTGTCCCACGATATGCCGCTGCAAGAAAGAAAACACGTTAACGCTTGTCTAGGCCATCTAGAGTTTGCTGTCACCTAACATGCTTCTCCAGGCAGAAGACTA  
AGGAAGTTCTTCGCCCCATACACGAGTTGGCAGACGAACCTGAAGGCCAGGGCTGGCGCGTGAGCCCACGCGGGTACAGACCGCTCATGTTAGGGGAA  
CTACAACGGAAAAACGCTCGGAAATTCCCTGCCCTGAGCATTTCCACAATATACACTGTTGGATTCTCGCTGCGAGCAAGCTCGCGCATTTTGCCTTATTGGCC  
GTCACTAAAGTTGTACAAATGATAGTTAAATAATCCTGTTGATCTGTATTGTTATCTTCACTAACAGCTAACGGTATTTGACAGCGGACCACGGG  
AGCACTCACATATTGGAGAACGAAAGAACACCGAAATTAAAGTGGTGGATGGCATTAGACTTGAACGTCCACGTTGAAGGCCCTGTTCATCCGGAT  
GTAGAGCTTGTCAAGCGGAATCGAGTACATCCAAGAGACTTGGGCGGCTCATCCAGAACCATGCTGAAGCAATAACCACCGATGTTGATTCTCTAGCTGGGAG  
CGTGTGCGCTGGATCTGAATGTCCTGCAGCATATTGCGCAGCACGGATTGCTGTAAGCAAAAACAAGATATTATTGAATGTCAACTTAACCGCAATCATATA  
AGGGTTCACACTTGCCTCATAGAGCTGATAGAGCTTCACTGTTCCAGCTGCTGAGTTAACAAATGGACTGGCATTGTTGATTCTATTCTATTTCACAAGTG  
CTTGAATTCTCGCTAACGATGACGACGGGAAGGAGTGCACCGTCTAACCGCTAGATAAGAGTTGTTCTGCTCTCAACTGAAAGTGAACCTCGAACCGAC  
TTTGACGTATGTCGCTTTATGAAATTGCAAGGCAGCGGCTGAGTCACGCAAGGAATGCCGTTCCCTCACGCTTCTATGCTCTTGTGTTGCTAGTAGCACA  
ACGCATGCTCAGTTGGTTCTGGACGGCGGTGAAATACATGCGTTTCGCGATGTTGCTCGATAGTGTCCGGTACTTGACAGGTCCTCTAGATGGCATCAGTTGGGTA  
GTTGAAAACAATTAAAGCAGATTTAATGCTGTGAAAGTGCCTCACAGTTGTTAATTGTCAGAGAAAATGGACTTCACCAGCGACTACCGCATCGCGTAT  
GGTGAATTCTGACGATCATACTGATAGGCTTATGACCGTCTCGGACTCCCTGCCAATCGATATCGGGCGGCGTGTGAAAGATTCGCTCTCAAAAGGCAAAT  
CTGGCCTTGTCTGCTGTGGCAATTGCAATTGCTTACGGGGCGAAATCTACAAGGAGTACCGAGGAGGGTCAGATCAACCTGAAGGACGCCACACTCTGT  
ACAGCTATATGAACATTACGGTGGCTGTTATTAAACTATGTCGCAATGATAATCAGTACCGATGTCGGCAAGGTGTTGAGCAAAAGTGGCTTCTTGTATACCTAA  
AAGAATTCCGTCGGACAGCAGGTCGCTGTACATATCCATCGTTGGCTCTGGTCAAGACCGTGGCTTCTCAACAAATTGAAGTGGCTTCTACTGCAACAGAG  
GGCGCAGCATCCCAGAGATGAGCTTGTACCGGCTGTTCCCTTAATTATTGCAATTCTCAATAACTGCTACTTGGCGCAATGGTGGTGTGAAG  
GAGATTCTGTACGCTCTGAACAGACGGCTGGAAGCGCAGCTGCAGGGAGGTGAATCTGCTGAGAGGAAGGACCAAGCTAAAGTTGTAACACTAAATACCGCATGCA  
GCGATTTCGCGCTTGGGGATGAACCTGACCGACTGGCGTATCGCTATAGGTTGATATATGTCATTGCGGAAAGTATCTGACCCCAATGTCCTGTCATGATTCTG  
TCGCTCATATGCCACCTGCTCGGAATAACGGTGGTTCTACAGTCTGTAATGCCCAGCGAACCTTAATCATGGCAAGCCGTACGATGGCTTGGATCGCTGA  
TCAATCTGGTTCTCTCCATCTCGCTGGCGAGATCACATTGCTCACGCAATTGTCAGGCTTACTCTGCTGGTACGGTACCGAAGTACGTCATTCTCAGGAGATG  
AATCTCCAGCATCGGAACAGCCGCTACCGTCAGGCACTCCACGGTTACTCTGCTGGTACCGTACCGAAGTACCAAAATTAAACCTTGGCTTGTACGAGCTGGACA  
TGGCAGCTGATCAGCAATGTCCTCTCGGGTGGCCAGCTTCTGCTGATCTCGTGCAGGGCATCTGCTCCAGGGCTTCAAGATGCAATTAGCTAATCGATGTTACCCAC  
CTGGCTGAAGTGCATCAGATTCCGGACTGCGGGAAATAATTAAATAGTTAGTAAGCTATAGCTTGCACATTAAAGCACAACCTCACGTTAAATCCCTGCAAGAA  
GGCCATGGGTTCCGTGGTCTGGAAATAAAAAACAAATAATAGTTAGCAACCAGGGAAACCCACATGTGACCGACGTACTACAATTCCGATCCGATACCT  
CCACCGTTTCCGGAATATCCCTTGTGATATGACCTCCGTGGAGTCATCTCGGAATCAGTGTAAAGTGTAAAGGGAAATAAAACTTAGTGGCAAACCTGCGCCTT  
GGCTGGATAAACAAACACAAAGTGGCGCGCAATGATGCAAGTACAAGCGTGTGCTACATGCAAGGTTCTACTGTCATCAATTGTTGGCGCACTG  
TACCTTCTTGTGCCACGACATTGGCTGTGATATACATCTAGTCGGCTATATCTGATCAAGCGATCTGTTGGGAAAGGAGGCTAAATATTAAAAACTCGCTAAC  
AAAAAAACTACAGAATCGTAGTGGCCACCGACTATTGGATTGGCATTGCCACGCAGCGCAACATGCCGTAGCCAATGTTATCGATTGGAGCGCGTGTGTTG  
AATTGCTCACCCCTTGTGGCAGACGAATTGTTGCTGATGCTTAATTGTTGTTATTGCCATCGGAATTATTGTTGGCAAAATTCAGATTGTCGTAACGTCATCAT  
GTTGTCGTTGTTGTCGCCCTCCGTAGGCTGGATCCACAGGGCTAGCCAGGGCATATTGTCGTTGCAAAACTCTGCGCAAGTGAAGGCTCCGTTGCTG  
GAGACCGAAGTGCCTGACTCTGCAAGAAACTTCGTTGCCATATCCAAAATCCAGGAACCTCTGAAAGCTTCAAAATTGTCGACTATTGCTACGCAAAACAAATC  
GAAGTGTGTTCTGCAAAAGGGCAACATTATTGTTGGGTAAGACCGAATAAAATACATGTCGCTCACGTTGTTAAAGTCAAGCTAACGACCGGGACTTGTGAAGA

# Chromosome 3R: 23,049,569-23,054,170

Drag>Select:

Save

Chromosome bands.

Fly\_gold\_cDNA.

Fly\_cDNA.

FlyBase\_feature.

Contigs.

FlyBase\_feature.

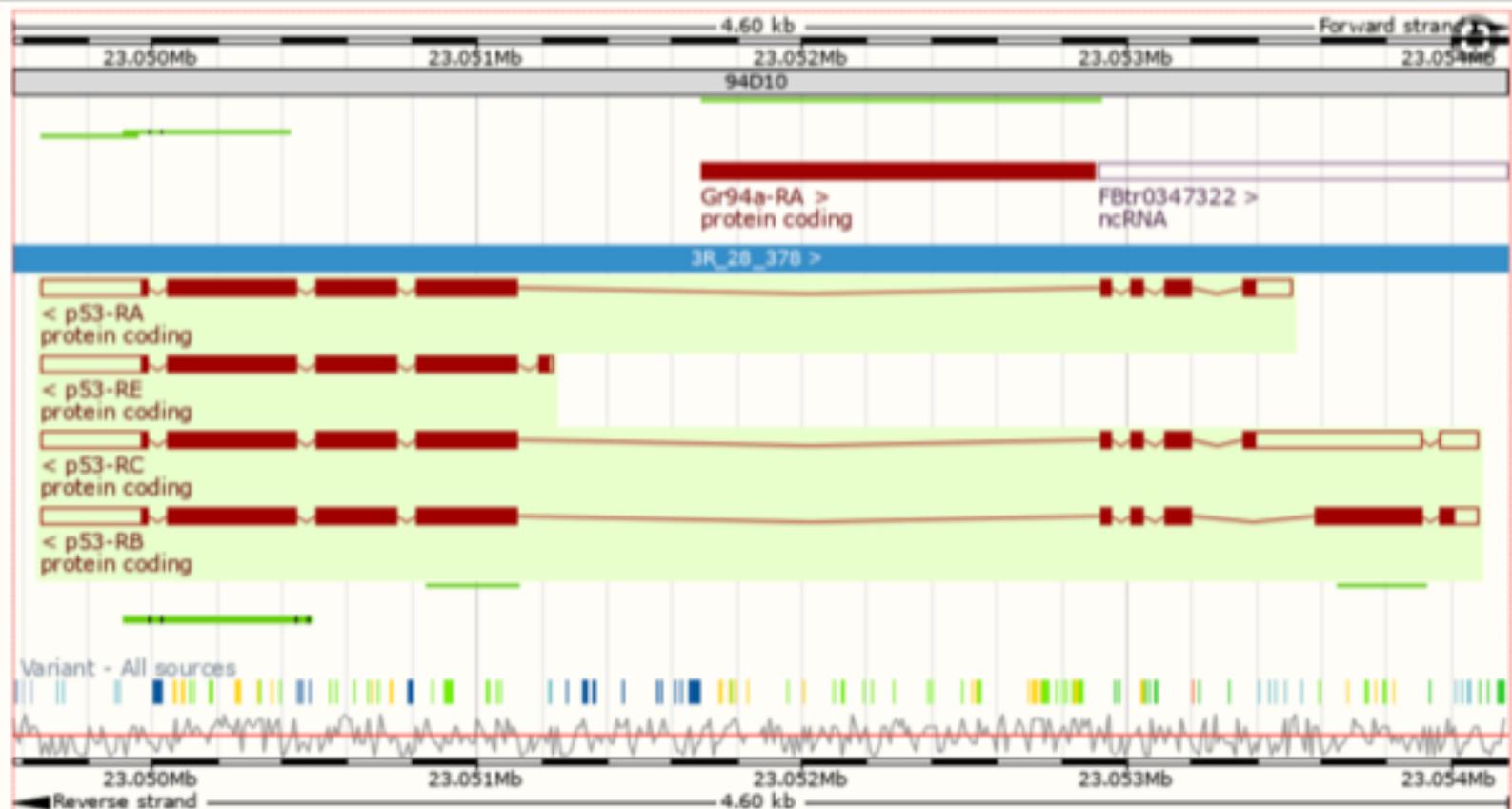
Fly\_cDNA.

Fly\_gold\_cDNA.

Variant - All sources.

%GC.

Variant Legend



- Variant Legend
- stop gained
  - splice region variant
  - 5 prime UTR variant
  - non coding transcript exon variant
  - upstream gene variant

- missense variant
- synonymous variant
- 3 prime UTR variant
- intron variant

Gene Legend

- Protein Coding
- Ensembl protein coding

- Non-Protein Coding
- RNA gene

There are currently 22 tracks turned off.

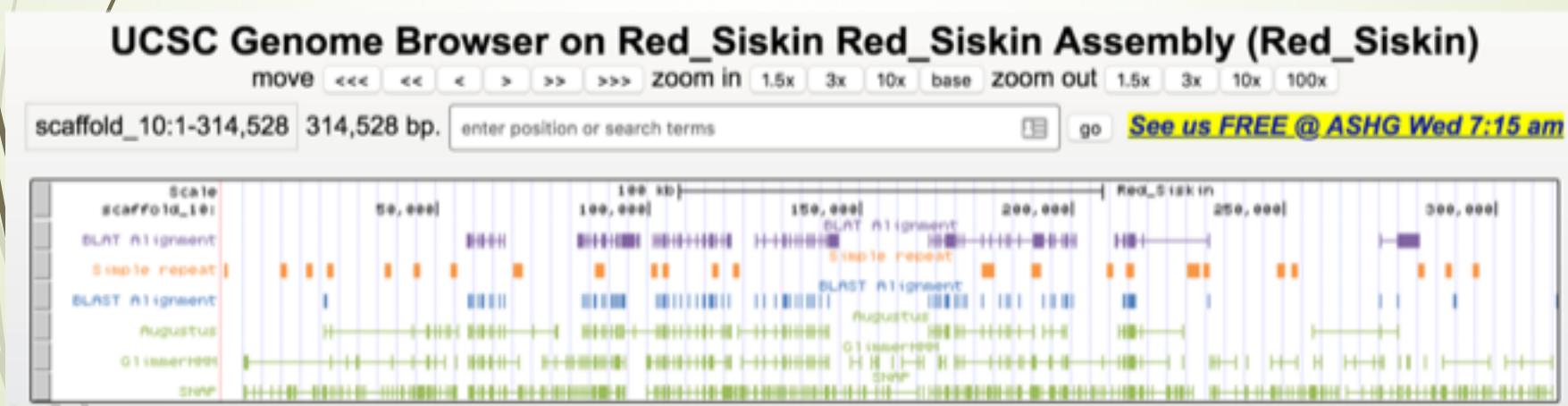
Ensembl Drosophila melanogaster version 94.6 (BDGP6) Chromosome 3R: 23,049,569 - 23,054,170

# Red siskin genome browser

- ▶ Multiple sources of evidence:
  - ▶ BLAT alignment
  - ▶ Repetitive regions
  - ▶ BLAST alignment
  - ▶ Gene annotator and predictors



 **Galaxy**  
PROJECT

The Galaxy logo consists of a stylized yellow and grey equals sign icon followed by the word "Galaxy" in a large, bold, sans-serif font, and "PROJECT" in a smaller, lighter font below it.

# Genome Annotation

Augustus

SNAP

GlimmerHMM

Genemark-ES

FGenesh

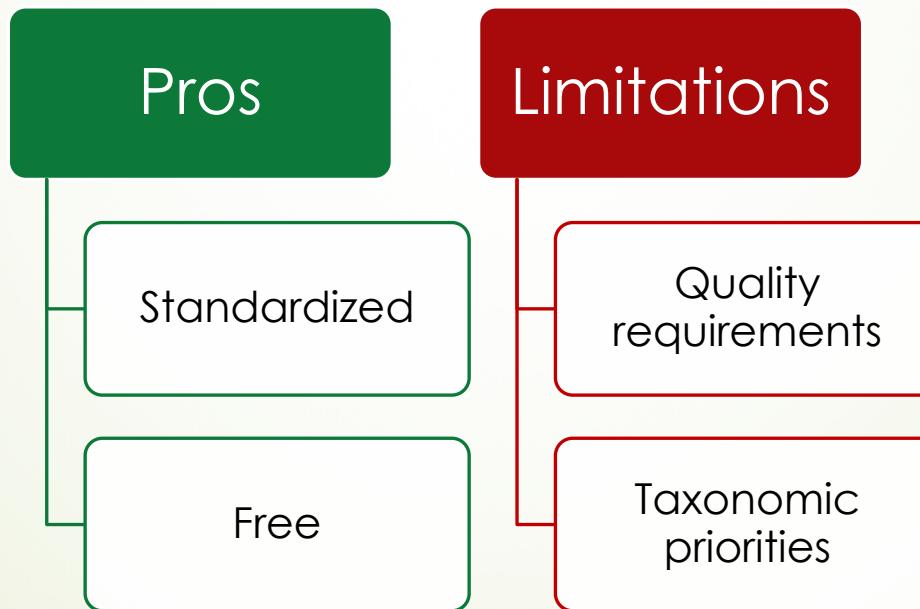
Gnomon

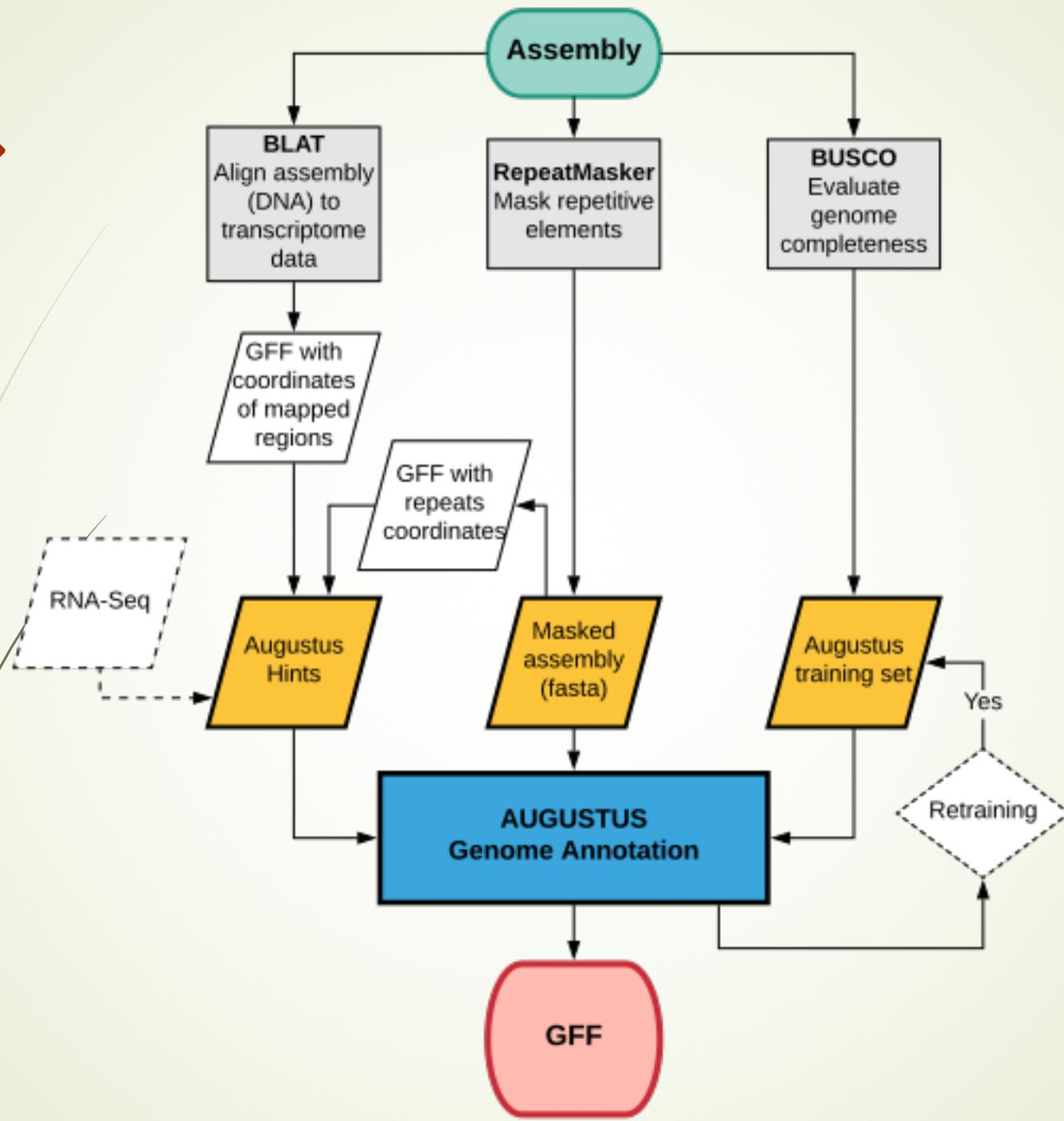


MAKER  
Annotate this!

Web  Apollo

# Other alternatives





# Folder structure

- ▶ Create the following folders:
  - ▶ assembly
  - ▶ busco \*\*\*
  - ▶ repeatmasker
  - ▶ augustus \*\*\*
  - ▶ blast
- ▶ Each job will be executed from its specific folder.

**You should have  
your augustus  
and busco  
folders**

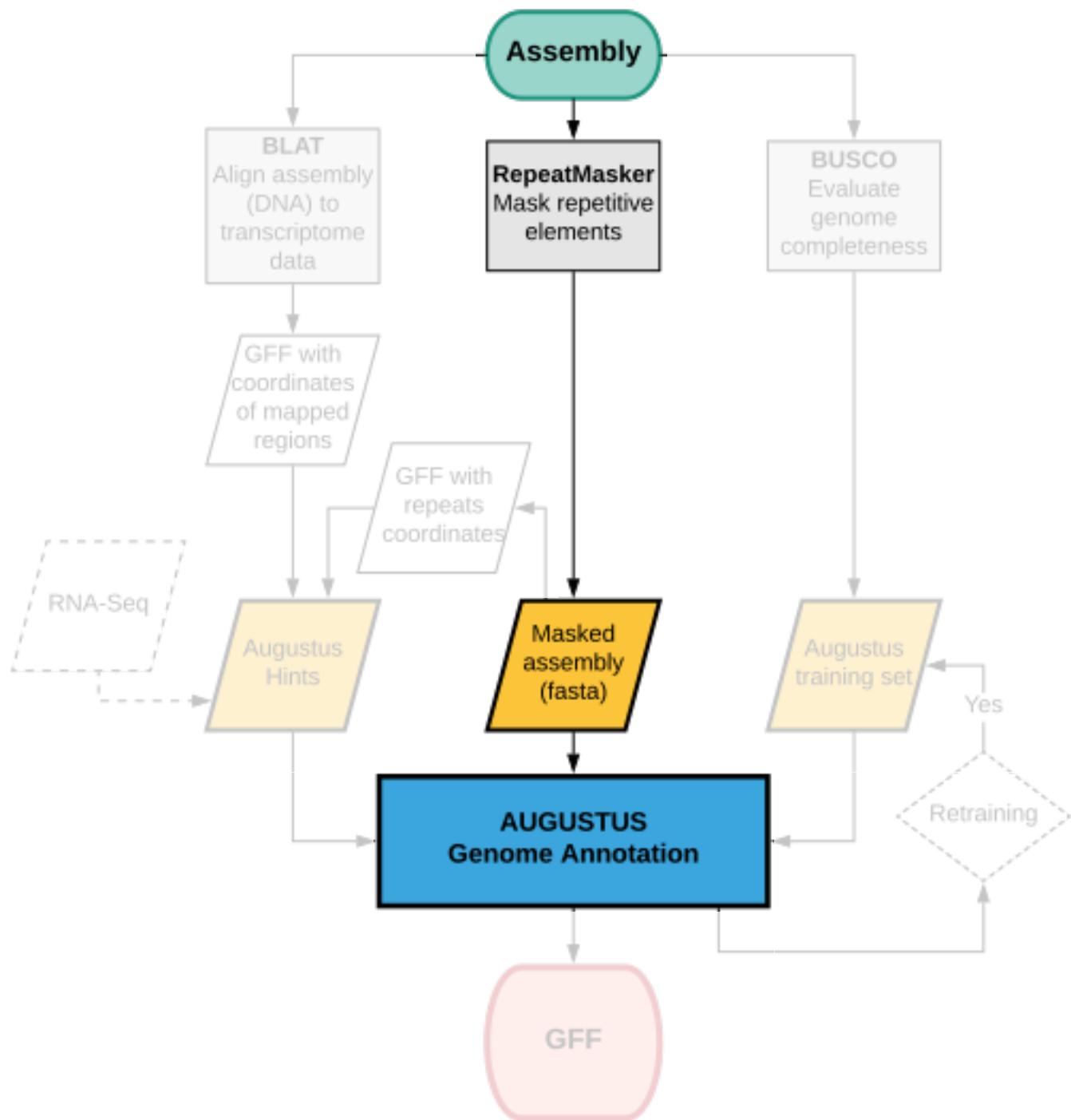
\*\*\*

**Why?  
It is easier to find everything later.**

# Data prep

## ► ASSEMBLY:

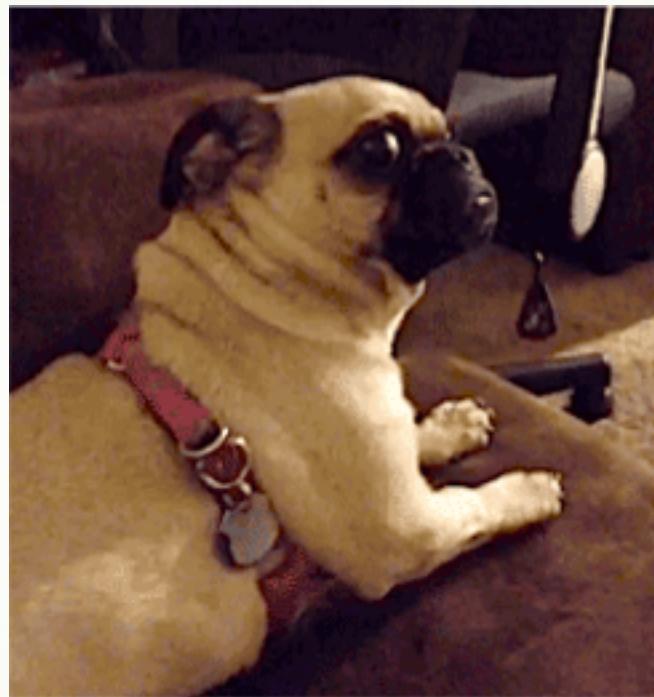
- cd to your **assembly folder** and copy the same config you used as reference for the variant calling pipeline



# RepeatMasker

**RepeatMasker is a program that screens DNA sequences for interspersed repeats and low complexity DNA sequences.**

(from the RepeatMasker website)



&gt;Contig3141\_pilon

GACGGTCTGTCTAGTCAGCACTCATCCACTGTTCCAGCTGGCAGCCGCTGAAGGTGAAAAAATCTATAGATAGATATGTAGATGTAAAAGCAGATGAGAACCGGTCGCATATATA  
 GGAGAGTGAGCCCTCGCCTCCAGAGATGCTCTTAGAGACTCCGCCCTGTATACTGTGTCTAGCCATATAGACAACCAACGAGTCACCCACTACCGAAAAGTGCCTGGACTGAACG  
 TCTCGAGAACAGCTAGCACCAAAATAAGGACCATCCGCACATCGAGCACCTATGGAGGCTATCTCACCCCGTGGCGAACACTCAGCGCTCACACACTATGCGTAAGCCTCTGATT  
 CCAAAACAGCATTGTTGAGAACTTGCCCTACTTCTACAGATTATCGCTCAAATCTAACATATGCTGAGGTTCAATCTGCCCTCCGCTACTTTAGTGGAGACACATGTAATAACAGTCAG  
 AGAGACTGAGTCTACAGctctctctccctctatcattGCTATATGCGAGTGCTGAGGAGACTAGCTAGTGACACGTGCTGGTATCTCTATGCTCTCATTGCTAGATGGAG  
 GAGAGACGCATACTTAGCTGAACGTCCTCATTACTCCCTACATCAGATAGCATAACATCTAGAAGTTACTGTGGTAGCACC [AtttctatTTatTTTTTTatTTTTTT  
 AGCACTGAGCGCGAAAACAGCTATTTGAGATATTCCATCGAAAATAGACTCCAAAACATCATGAAGATGACCACTTCCCCCTGATATAAGGATTACAAAGTTAAGATGGAAGAT  
 GCTGCTTCCCTTCTCGCCCTGGTAGCAGCCGACGGCTGGATTGGGCTCCACATTGCAACCCCAGGTCTGGCTGAAGGACAAGAGAGAGATACTTGGCAAAAATCCACGGTTGG  
 TTGACCTTGTATGACTTCTGAGGTATATAGATTCAACGGATGGCGATGACACACTCTCATCTGCAAGAGAGCAGAACCATGGCGAGGACTGCTTGGAACTACTTACTGCCAGAGAG  
 ACAGGGACTCTCCTGGTAGACGCAGGTTCCAGAAGGTCCCCATTGCTCTCTCTCAGGGAGTCTGCTATGTGCTTATCCGAGATTAGAAGATACAAACATATTGAGTCTCCGCT  
 CCGTTGTATATCGTATGCCAAGGAATTCTGTTATTACTCGAGACCTCACAAAGATTCAAGTTCAAGCACACTTTATTCTACTCTCATATGGATTCACTGTGATTTGATTAG  
 GTTGTGTTCTCTCAAGGGCAGCCCTAGTCTGCAGCCTTGCTGATGAGGAGAGACCTCAACAGGGCGTATTGGCTGCCTACTATTGATGGAGAACCATGACTATTATGGCTA  
 TTTCTTAATGGTCGGCCCACCCATAGAGCCGCAACCGGAAAGCCCTACTGCTGGATTGACGCAAGAAGGTATAAGTAGCCCTCAGCTTGTGATTGACATTATTTGTCCTCC  
 CCACGCCAGCACTGGATGGGCTAATTGCTTAGACAGATACTGGGAAGCGCGGATCACCATGGCTGGTAATTAAAAAAAGCTAACACACCAACTCAAAGTTACTAGTGGCCATGGATC  
 TTTTCACGACAATGGCATTCTCTAGCCAACACCTCAGTGATTACTCTACTTTCTCTCCCTCTGCTGGGGAATAGTCAATGCCCTCGGCCAACATCCATATATCG  
 GAGTCCTGCGAGTAACGTCTATGATCTTGGAGCGTACTACACAGATCGGCCAACAGGCAATTCTACCTGACTCTTGGCAACAGGAAACATTAAACAGGTGATGGCTAGGGTTG  
 GTTCCCACCTGAGCCACAAGCCATGTTGGACCACTTGTAGCCCTAGCCCTAGCCCTAGACCAGCTTGTCTCAACCCCTTCACAAAGTGGGACTGACATGACTGGCAGTCTGG  
 AAGCAGGCAGGCAGGCATTGGCACAGCAGGGTAGGGTTGCTTNTAGAGCTACTCAGCCACCTCACTACCCGTGCTGTAGTCTCTCCCTGCTGTGACGGCAG  
 TAGTGTACCCACCTTCATAGAAGACTGCGGGCGTGGCTGGCAGAGTCGCGCAGCAGTCTGCTTCTCCCCAGTACGCAAGGTGGGCAAGAGTACACAATCTTACATAACAGGT  
 GAGTTATCGTCTCCATCTACACTGACTGCTGCTAGCACCAAGGAATCAAGCCATTGTGCGCTCTGCCCTCATCTGCTATACTCCAGGCATCGAGCGCTCTGCTGGCCTGCA  
 ATGCGTTAAAGATTATTAAAGTTATCGGTGTTGCTGACATCTGAGTTAAAGGTTGTGCAAAGCCATCCCACAGCCCTCAAGCACAATACTGTAGGCAAGGCTCTGGCTG  
 TCCCTGAGGCATGTGAGGTTGCTCCCTCCCCAGGTACAAGGACTGAGACTTTCCCAAGGCGTGGTCAAGACAGAACTGGGTGTTCTGAGACTCACGATGATATGCAAAACGA  
 ACTGTTGCAAAGCAGGTGTTCCATGCTGGAGGTGGGGACACAGAGCTTGTCCACAGCCAGGCATGGGCGCCCTAGGCTGAGCTGTGAACTTGAACAATTAAACAGTCAGAGCTG  
 ATGGATGGTAA [AtttctatTTatTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTTatTTTTTT  
 CCAAAATTTATTTGGTCAAACACTCAGATTGCTGGCTGCTGATTGTGAAATCCATCTTATTTTCTCCCTGTCATTGCTTACATGCCATGCAAGTCCCTCTGAGCTGCTC  
 GGCAGAGCCCCCCCCGGCTCCCTGGGGTGAGCACATTCTTGCTGAGCAGGAGAACAGGAGCTTCCCTGGCTGATGATTCTGACTAACTGTGGAGGAAGTTAA  
 TTTTTATTTAAACACTTTAAAGACAGCTATTGTTGACTGTAAACAGCAGCTTGTGCTGCTGTTAAATGTTCTAGTCTAAATGAAACATACCCCTGAGGTATGACTGAGTC  
 TCCCTGGGGAAACTCTATTGCAAGGGTCAGTGCACTGGCTTGTCTAGTGTCTTAAATGTTCTAGTCTAAATGAAACATACCCCTGAGGTATGACTGAGTC  
 GTTTGGGGCCCCAAGGGGCTGGCACAGCCCTGTGGGTGAGGCTGCTCATCAGAGGAAGCAGCAGCAGCAGGAGGGCAGGAGAGGGTCACTGTCTGGCTAGTGGCAG  
 CAGCCCTGGCTGGCATGCTCAGGAGCTCAGGTGTGAGGGGGACAGCTTGGTCACTGTCTGGCTAGTGGCAGGGATCTCAGCTGGATGCCACTGCAGTGGGGCACACA  
 TCTCCCTTGTGCTGATAAGCTGGCTCTTCCACATTATGCTAAACAGAGACATGGAAGGTTATGGAGATCTCTGGATTAAATGAAATTGTGTTGCTCAATCTCCATGCT  
 GGCTGTATAAAAGGTTGCTTAAAGCTTGTCTGAAATTGATCTGGAAATTAAAGGAAAAAGCTTATTTCCCTTGTGACAGCAGCAGGAGGGTCACTGTCTGGCT  
 TGTTGTAACTCTACAAACACTCTGTGGCTTCCCAGAGAAAACCAAGATGCTGGCAGAGTGACAGTCCTGGAGGAGGTGATAGAAAGGATTCTCATGGGGGGGTCTGT  
 GTCTCCCCCATGCTGCTGCTAAGAATGCTAAGAAGTGACAGCAGCAGTGGCAAGGCCCTGTGAAGATAGCAGCATAGTCAGAGGCAACTCCAAAGAAGTGA  
 AAACGTGAACTGCAACAACCACCCAAACACATGCCATGGAAACACCCCAACAAATGCTCAAGCTTAAAGGCTCTAGTGTGAGGCCACAGAGAGCAAGGGCTTT  
 AGCTGTCAGAGCAGGATGTTGGGCTGTTAACATAAGCTTATGCTAAACATGAGATTGAAAAGAAAATCAAAGGAGAACAAATCAGGAGACTGAAATGAAAGGTTGG  
 GGATCAGCTCCCCAGGTGGGAATGGGCTGAGCCTCTGCCAGTGCTTGTGCTGGATGTTGACCCCTGGCTCCAGTGTGCTCCCATAAATGTTACCTGATAGCTGAG  
 ACTGAAGAGGAGCTTGGATACTGTTGATGGGGAAATGCAAGTCTTAAGCAGCTGCTAAACACAGTGTGAAGCCTGCTAAGTGTCTAGCAAAGTCAGAACTA  
 GCCTGGGGATGTCAGTGTCTGCAAAGCCACACAAAAATGGAAGTAAAGTATGGCTCTGTGCTGGAGAGAAATGAGATATCATCACTGTGTTG  
 GAAGGAATGCCCTGTTCTAAGGAGCTACTGAGCTTCTACCTAAAAAGCTGTGAGCTTACCTAAAAAGATCAAAGCAATTGACTTTAACCGAGAGTTGAAGAGACAA  
 TCT [tagaaaaaaaggccatccccactggacacgcctctgttaaggagttgtcacagcactcagccgtccagagctcaagaagagttggacaatgcttc  
 gctgacccaggatggggatggggatggggatggggatggggatggggatggggatggggatggggatggggatggggatggggatggggatggggatggggatgg  
 siskin\_Contig3141\_pilon.fasta.masked

Run information: input file, total length, % bases masked

Types of repetitive elements, quantity, length and percentage of sequence

file name: siskin\_Contig3141\_pilon.fasta  
sequences: 1  
total length: 5638391 bp (5638313 bp excl N/X-runs)  
GC level: 41.23 %  
bases masked: 196516 bp ( 3.49 %)

	number of elements*	length occupied	percentage of sequence
--	---------------------	-----------------	------------------------

Retroelements	488	122957 bp	2.18 %
SINEs:	32	2436 bp	0.04 %
Penelope	1	174 bp	0.00 %
LINEs:	414	101867 bp	1.81 %
CRE/SLACS	0	0 bp	0.00 %
L2/CR1/Rex	413	101693 bp	1.80 %
R1/LOA/Jockey	0	0 bp	0.00 %
R2/R4/NeSL	0	0 bp	0.00 %
RTE/Bov-B	0	0 bp	0.00 %
L1/CIN4	0	0 bp	0.00 %
LTR elements:	52	18654 bp	0.33 %
BEL/Pao	0	0 bp	0.00 %
Tyl/Copia	0	0 bp	0.00 %
Gypsy/DIERS1	0	0 bp	0.00 %
Retroviral	52	18654 bp	0.33 %
DNA transposons	77	11689 bp	0.20 %
hobo-Activator	18	1779 bp	0.03 %
Tcl-1S63B-Pogo	5	1664 bp	0.02 %
En-Spm	0	0 bp	0.00 %
MuDR-15985	0	0 bp	0.00 %
PiggyBac	0	0 bp	0.00 %
Tourist/Harbinger	15	1083 bp	0.02 %
Other (Mirage, P-element, Transib)	0	0 bp	0.00 %

Rolling-circles	0	0 bp	0.00 %
-----------------	---	------	--------

Unclassified:	24	3841 bp	0.07 %
---------------	----	---------	--------

Total interspersed repeats:		137887 bp	2.45 %
-----------------------------	--	-----------	--------

Small RNA:	18	1075 bp	0.02 %
------------	----	---------	--------

Satellites:	1	62 bp	0.00 %
-------------	---	-------	--------

Simple repeats:	1000	48623 bp	0.86 %
-----------------	------	----------	--------

Low complexity:	196	9738 bp	0.17 %
-----------------	-----	---------	--------

\* most repeats fragmented by insertions or deletions have been counted as one element

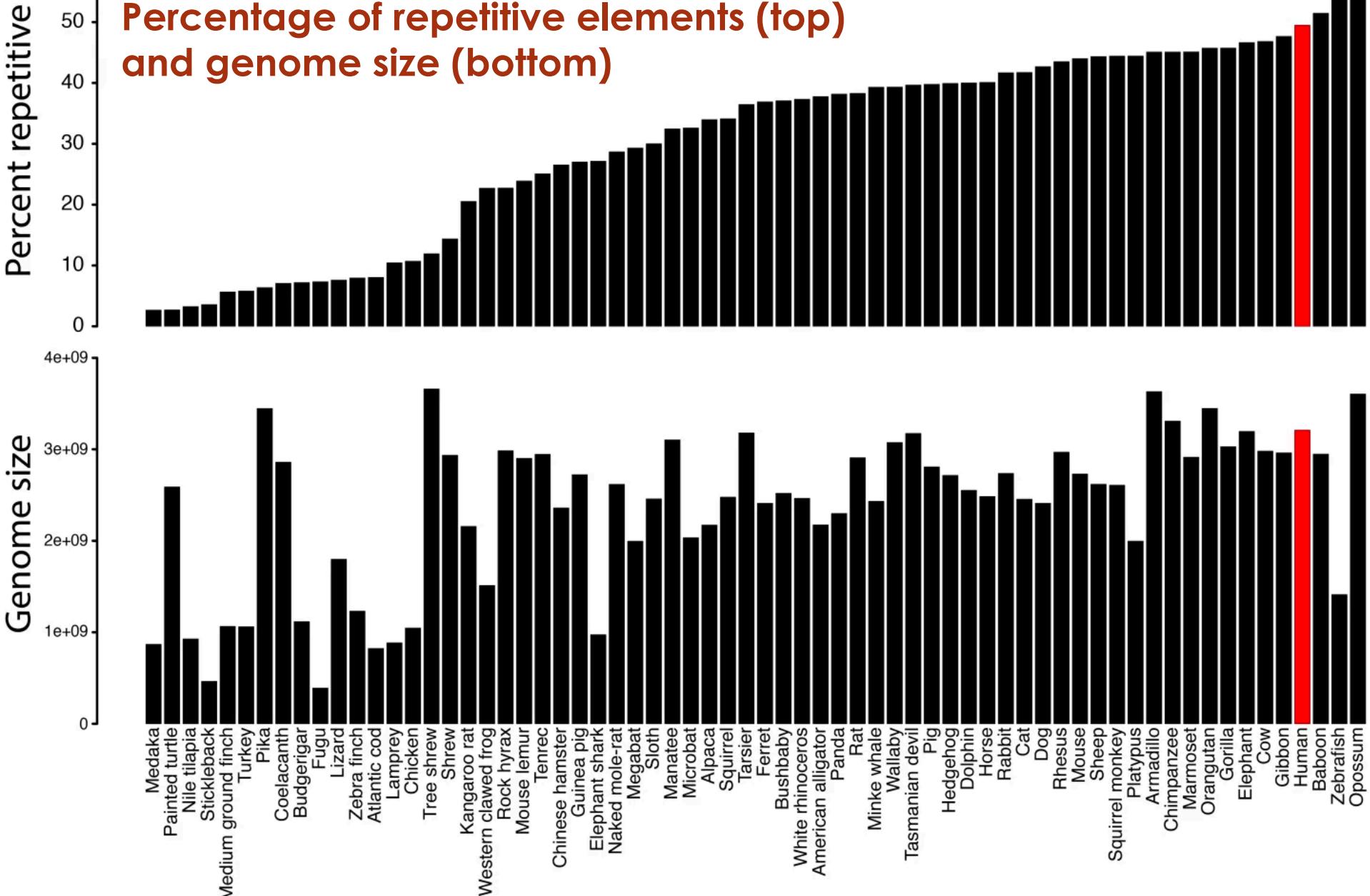
The query species was assumed to be gallus gallus  
RepeatMasker version open-4.0.6 , default mode

run with cross\_match version 0.990329  
RepBase Update 20150807, RM database version 20150807  
[tsuchiya@login-30-1 repmasker]\$

[tsuchiya@login-30-1 repmasker]\$ head -n 200 *.out													
SW score	perc div.	perc del.	perc ins.	query sequence	begin	end	position in query (left)	matching repeat	repeat class/family	position in repeat begin	end	(left)	ID
17	8.8	0.0	0.0	Contig3141_pilon	684	627	(5637764)	+ (CT)n	Simple_repeat	1	24	(0)	1
26	11.3	0.0	0.0	Contig3141_pilon	819	856	(5637535)	+ (T)n	Simple_repeat	1	38	(0)	2
38	20.8	3.4	2.2	Contig3141_pilon	3372	3459	(5634932)	+ (TGTT)n	Simple_repeat	1	89	(0)	3
12	8.2	3.6	3.6	Contig3141_pilon	3852	3879	(5634512)	+ (GCCT)n	Simple_repeat	1	28	(0)	4
788	23.1	5.7	2.9	Contig3141_pilon	5845	6049	(5632342)	+ CR1-X2	LINE/CRI	3923	4133	(6)	5
18	18.1	0.0	0.0	Contig3141_pilon	6253	6298	(5632181)	+ (ATT)n	Simple_repeat	1	38	(0)	6
14	21.8	0.0	4.5	Contig3141_pilon	17123	17168	(5621223)	+ G-rich	Low_complexity	1	44	(0)	7
459	24.3	0.0	1.8	Contig3141_pilon	23515	23627	(5614764)	+ CR1-X1	LINE/CRI	4823	4133	(4)	8
1488	27.5	1.5	1.2	Contig3141_pilon	28192	28600	(5609791)	C CR1-F2	LINE/CRI	(16)	4497	4688	9
254	24.1	0.0	0.0	Contig3141_pilon	28926	28983	(5609488)	C UCON24	Unknown	(111)	263	295	10
20	22.2	0.0	3.4	Contig3141_pilon	30324	30384	(5608007)	+ A-rich	Low_complexity	1	59	(0)	11
21	0.0	0.0	0.0	Contig3141_pilon	40281	40303	(5598088)	+ (T)n	Simple_repeat	1	23	(0)	12
969	24.0	2.5	5.5	Contig3141_pilon	42725	43050	(5595341)	C CR1-X2	LINE/CRI	(41)	4098	3783	13
15	8.0	0.0	3.6	Contig3141_pilon	44280	44308	(5594083)	+ (TCTTC)n	Simple_repeat	1	28	(0)	14
32	0.0	0.0	0.0	Contig3141_pilon	44314	44343	(5594048)	+ (T)n	Simple_repeat	1	38	(0)	15
13	18.4	7.0	0.0	Contig3141_pilon	44531	44573	(5593818)	+ (CTGCTG)n	Simple_repeat	1	46	(0)	16
13	18.0	0.0	3.0	Contig3141_pilon	47527	47560	(5598831)	+ (CCTCCC)n	Simple_repeat	1	33	(0)	17
631	24.5	6.6	4.2	Contig3141_pilon	51189	51380	(5587011)	+ CR1-H	LINE/CRI	4602	4798	(14)	18
19	0.0	0.0	3.7	Contig3141_pilon	52831	52858	(5585533)	+ (AAC)n	Simple_repeat	1	27	(0)	19
15	5.6	0.0	0.0	Contig3141_pilon	57134	57152	(5581239)	+ (T)n	Simple_repeat	1	19	(0)	20
1008	13.1	9.6	0.5	Contig3141_pilon	60288	60487	(5577904)	+ CR1-C4	LINE/CRI	4289	4508	(3)	21
16	0.0	0.0	0.0	Contig3141_pilon	64723	64739	(5573652)	+ (T)n	Simple_repeat	1	17	(0)	22
15	19.9	0.0	0.0	Contig3141_pilon	64868	64896	(5573495)	+ A-rich	Low_complexity	1	29	(0)	23
12	3.5	5.7	8.8	Contig3141_pilon	65872	65996	(5572485)	+ (CTTTA)n	Simple_repeat	1	34	(0)	24
47	0.0	0.0	0.0	Contig3141_pilon	72051	72093	(5566298)	+ (T)n	Simple_repeat	1	43	(0)	25
47	39.1	0.7	2.2	Contig3141_pilon	72913	73192	(5565199)	+ (GT)n	Simple_repeat	1	276	(0)	26
429	25.7	6.4	4.8	Contig3141_pilon	76071	76299	(5562092)	C GGLTR88	LTR/ERVL	(837)	234	2	27
38	2.3	0.0	0.0	Contig3141_pilon	78999	79042	(5559349)	+ (A)n	Simple_repeat	1	44	(0)	28
28	14.7	0.0	0.0	Contig3141_pilon	80244	80288	(5558103)	+ (A)n	Simple_repeat	1	45	(0)	29
35	18.5	0.0	0.0	Contig3141_pilon	80590	80640	(5557751)	+ (A)n	Simple_repeat	1	51	(0)	30
1021	23.5	6.7	1.7	Contig3141_pilon	84211	84687	(5553704)	+ CR1-C4	LINE/CRI	3956	4516	(27)	31
235	29.2	3.0	0.0	Contig3141_pilon	90620	90684	(5547707)	+ Chompy-2_Croc	DNA/PIF-Harbinger	6	72	(0)	32
13	9.8	0.0	0.0	Contig3141_pilon	92076	92097	(5546294)	+ (GGA)n	Simple_repeat	1	22	(0)	33
513	15.5	1.1	0.0	Contig3141_pilon	93425	93538	(5544853)	+ CR1-F2	LINE/CRI	3888	3982	(531)	34
12	8.1	3.6	3.6	Contig3141_pilon	97420	97447	(5548944)	+ GA-rich	Low_complexity	1	28	(0)	35
12	12.7	2.5	7.9	Contig3141_pilon	105002	105041	(5533350)	+ (TGTATA)n	Simple_repeat	1	38	(0)	36
17	12.5	0.0	0.0	Contig3141_pilon	109381	109406	(5528985)	+ (T)n	Simple_repeat	1	26	(0)	37
13	11.1	0.0	6.5	Contig3141_pilon	110496	110528	(5527863)	+ (TATGCA)n	Simple_repeat	1	31	(0)	38
263	28.0	9.1	1.0	Contig3141_pilon	123790	123890	(5514501)	C LFSINE_Vert	SINE/tRNA	(307)	152	43	39
26	10.0	0.0	0.0	Contig3141_pilon	124173	124204	(5514187)	+ (ATA)n	Simple_repeat	1	32	(0)	40
916	19.7	11.7	0.5	Contig3141_pilon	132695	132913	(5505478)	C CR1-C4	LINE/CRI	(7)	4504	4258	41
13	11.7	3.6	0.0	Contig3141_pilon	143379	143406	(5494985)	+ (CTGTG)n	Simple_repeat	1	29	(0)	42
13	0.0	0.0	0.0	Contig3141_pilon	146450	146466	(5491925)	+ (GCA)n	Simple_repeat	1	17	(0)	43
871	26.1	0.0	4.6	Contig3141_pilon	154617	154877	(5483514)	C MER126	DNA	(199)	250	2	44
15	19.9	0.0	0.0	Contig3141_pilon	155007	155035	(5483356)	+ A-rich	Low_complexity	1	29	(0)	45
20	16.4	0.0	0.0	Contig3141_pilon	158673	158706	(5479685)	+ (T)n	Simple_repeat	1	34	(0)	46
19	4.5	0.0	0.0	Contig3141_pilon	162947	162969	(5475422)	+ (T)n	Simple_repeat	1	23	(0)	47
19	0.0	0.0	0.0	Contig3141_pilon	163664	163684	(5474707)	+ (TA)n	Simple_repeat	1	21	(0)	48
311	21.1	3.1	6.9	Contig3141_pilon	170755	170856	(5467535)	C CRI-8_Crp	LINE/CRI	(813)	2045	1948	49
16	18.9	0.0	0.0	Contig3141_pilon	173844	173873	(5465318)	+ (AC)n	Simple_repeat	1	30	(0)	50
12	14.4	3.0	3.0	Contig3141_pilon	173908	173940	(5464451)	+ (AATGAA)n	Simple_repeat	1	33	(0)	51
400	18.4	0.9	0.2	Contig3141_pilon	175213	175287	(5463184)	C CR1-X1	LINE/CRI	(1)	4136	4071	52
14	27.1	0.0	0.0	Contig3141_pilon	176602	176645	(5461746)	+ A-rich	Low_complexity	1	44	(0)	53
12	7.8	3.2	6.7	Contig3141_pilon	187232	187262	(5451129)	+ (ATAA)n	Simple_repeat	1	30	(0)	54
12	26.5	0.0	5.8	Contig3141_pilon	191939	191993	(5446398)	+ (TTTC)n	Simple_repeat	1	52	(0)	55

## Vertebrates:

Percentage of repetitive elements (top)  
and genome size (bottom)



# RepeatMasker

- ▶ RepeatMasker has several repetitive elements databases (Eukaryotic species only)
- ▶ Commonly used species include:

mammal, carnivore, rodentia, rat, cow, pig, cat, dog, chicken, fugu, danio, "ciona intestinalis" drosophila, anopheles, elegans, diatoaea, artiodactyl, arabidopsis, rice, wheat, and maize

# RepeatMasker – job

- ▶ Create a symlink of the assembly in the repeatmasker folder:  
ln -s ..//assembly/myassembly.fasta .

- ▶ Parameters:
  - ▶ Short
  - ▶ 6 GB memory
  - ▶ Serial
  - ▶ Module: repeatmasker
  - ▶ Command:

## WHY?

**RepeatMasker**  
**saves all output files**  
**in the folder where**  
**the assembly is.**

**RepeatMasker -species chicken -xsmall -gff  
-pa \$NSLOTS assembly.fasta**

# RepeatMasker command

## **RepeatMasker**

**-species chicken:** RepBase species

**-xsmall :** soft-masking (repetitive elements are masked in low caps instead of replaced by N)

**-gff:** additional output in gff2 format

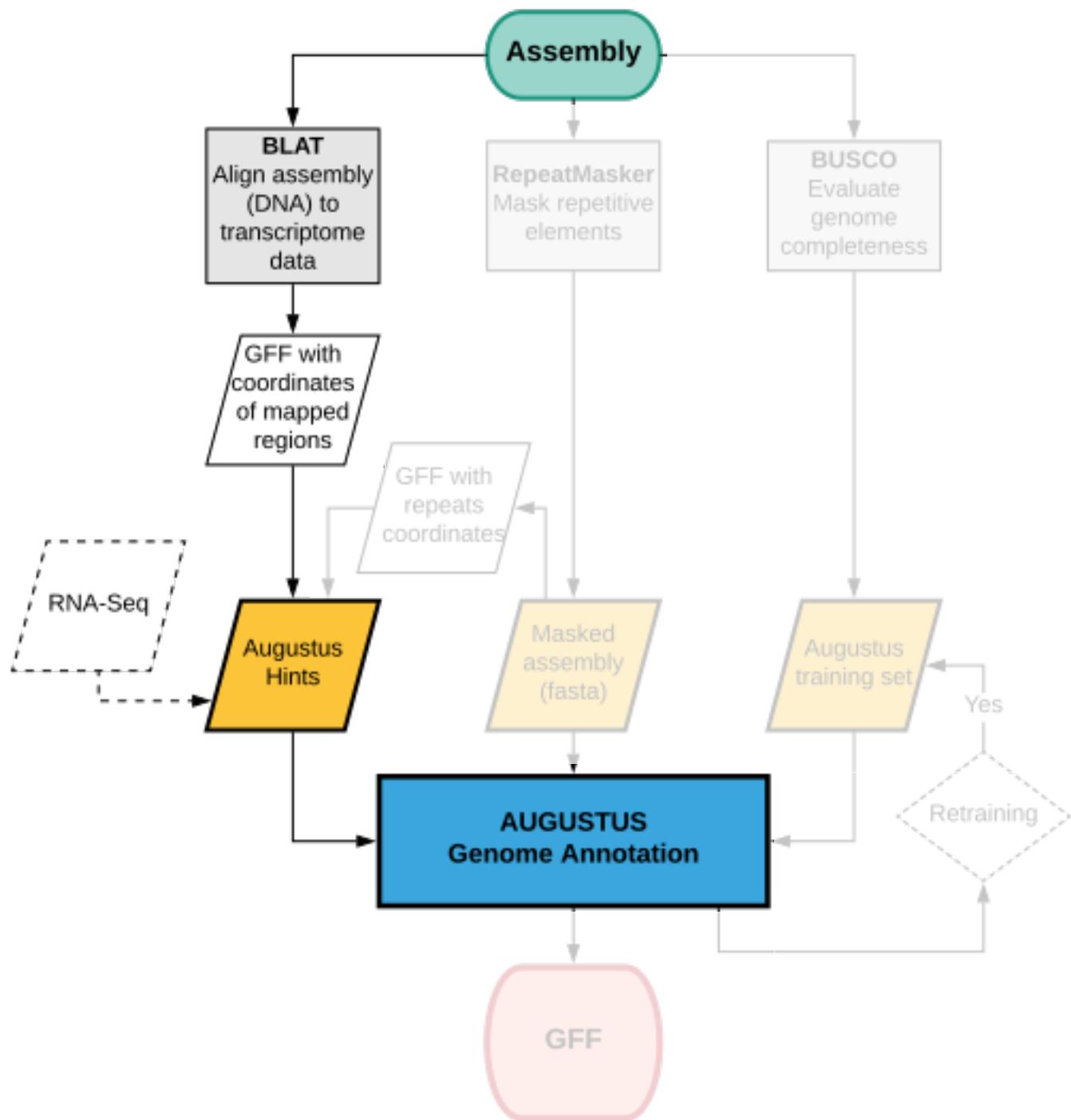
**-pa \$NSLOTS:** number of cpus

**assembly.fasta:** input file

# Other sources of evidence

- ▶ RNA-Seq
- ▶ Transcriptomes

**Today we will use the transcriptome of a different species to generate another source of information for the annotation**



# BLAT

- BLAST-like Alignment Tool

**TARGET**

DNA

DNAx

Protein

**DATABASE**

DNA

DNAx

RNA

RNAx

Protein

DNAx and RNAx correspond to 6-frame translated sequences



# BLAT - what does it tell us?

- ▶ It will provide information regarding exons and introns, based on the alignment of the transcriptome sequence to our assembly.

# Augustus – hints - Blat

- ▶ From the /pool/genomics/user/augustus folder:
  - ▶ Create blat folder and cd to it:  
mkdir blat && cd blat
  - ▶ Download the chicken transcript file (reference):

wget

[ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/002/315/GCF\\_000002315.5\\_GRCg6a/GCF\\_000002315.5\\_GRCg6a\\_rna.fna.gz](ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/002/315/GCF_000002315.5_GRCg6a/GCF_000002315.5_GRCg6a_rna.fna.gz)

- ▶ Unzip: gunzip  
[GCF\\_000002315.5\\_GRCg6a\\_rna.fna.gz](GCF_000002315.5_GRCg6a_rna.fna.gz)

# Augustus hints: Blat

- ▶ Create the job file using the Qsub generator
- ▶ Command:  
`blat -t=dna -q=rna \  
..../repmasker/your_assembly.masked \  
GCF_000002315.5_GRCg6a_rna.fna \  
siskin_blat.psl`



# BUSCO

- Benchmarking Universal Single-Copy Orthologs

What is a ortholog?

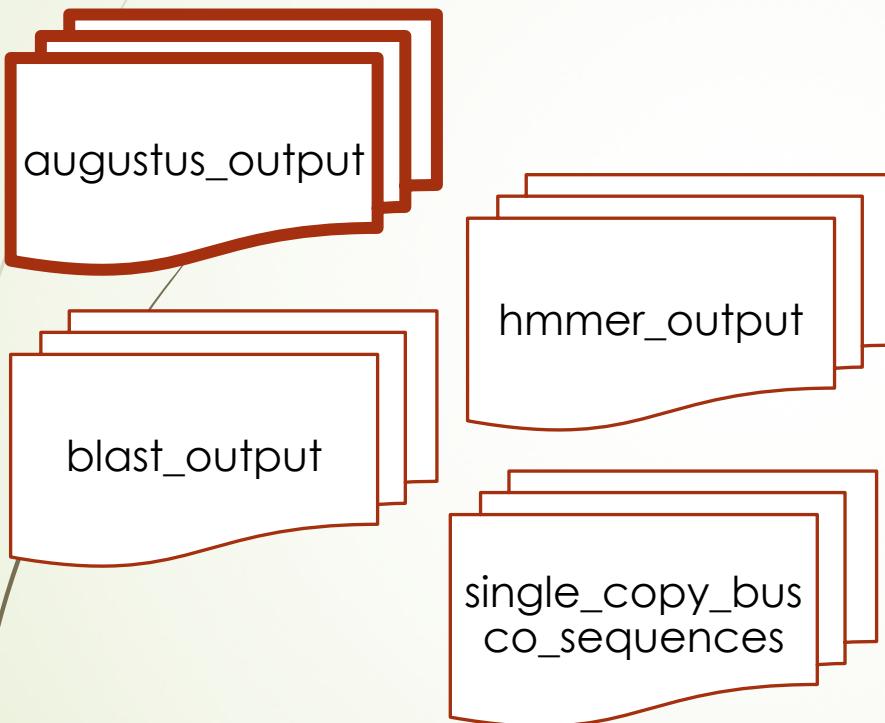
Orthologs are genes in different species that evolved from a common ancestral gene by speciation.

# BUSCO: What did we do?

- ▶ Input: red siskin assembly (full)
- ▶ Database: aves\_odb9
- ▶ Mode: genome

```
export AUGUSTUS_CONFIG_PATH="/pool/genomics/user_id/augustus/config"
#
run_BUSCO.py --long \
-o siskin \
-i /data/genomics/dikowr/SMSC/finished_assembly/masurca_siskin.fasta \
-l ./aves_odb9 \
-c 1 \
-m genome
```

# BUSCO Output - run\_siskin\_busco



short\_summary\_siskin  
.txt

missing\_busco\_list\_siskin.tsv

full\_table\_siskin.tsv

# BUSCO - results:

1. Check the results of your BUSCO run in the short\_summary\_siskin.txt
  - a. How many Complete (C), Duplicated (D), Fragmented (F) and Missing (M)?

C: 4610

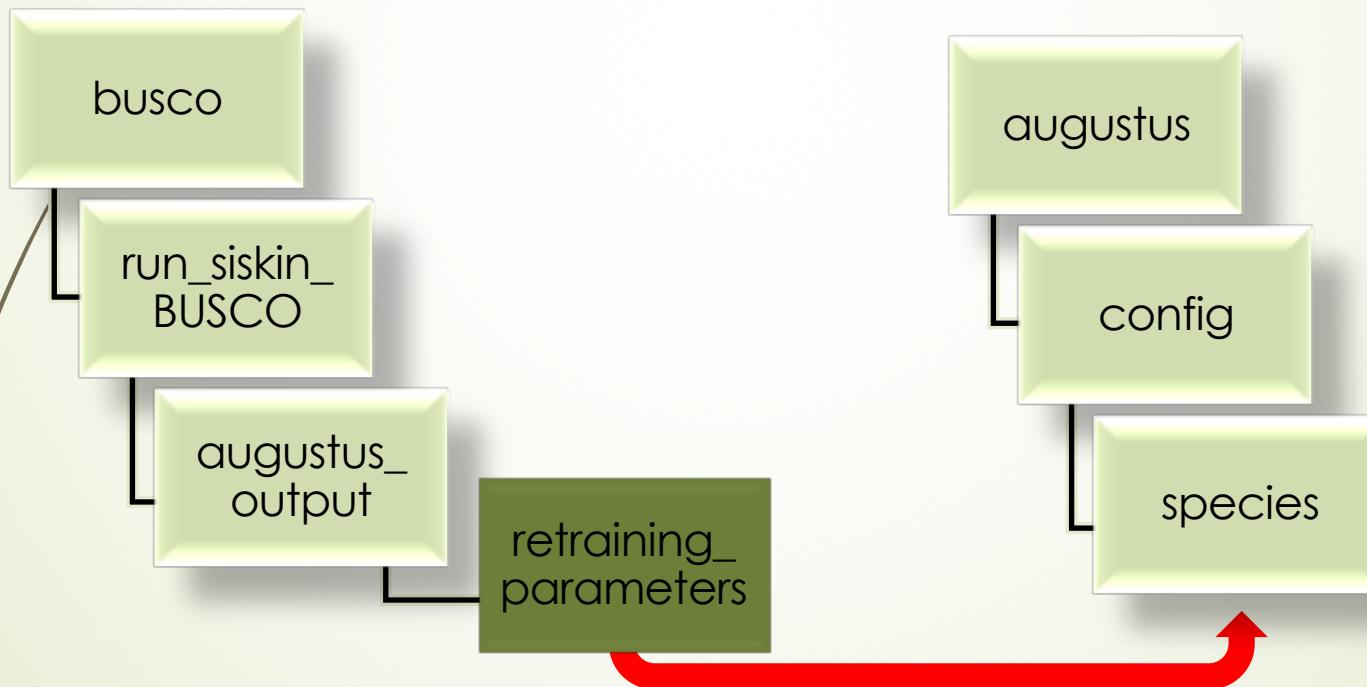
D: 63

F: 162

M: 143
  - a. Do you think this is a good or a bad assembly?

# From BUSCO to Augustus

- ▶ Copy the folder retraining\_parameters to augustus/config/species



# From BUSCO to Augustus

- ▶ cd to  
augustus/config/species/retraining\_parameters
- ▶ ls to check the names of all files:

```
BUSCO_siskin_busco_2684346740_metapars.cgp.cfg
BUSCO_siskin_busco_2684346740_metapars.utr.cfg
BUSCO_siskin_busco_2684346740_parameters.cfg
BUSCO_siskin_busco_2684346740_weightmatrix.txt
```



# Rename your retraining parameters folder

- ▶ Copy the prefix that appears in all files
- ▶ Go one level above (cd ..) and change the name of the retraining\_parameters folder to the prefix:

```
mv retraining_parameters  
BUSCO_siskin_busco_2684346740
```



# Creating hints for Augustus

# Augustus hints: Blat

- ▶ Convert blat results to hints  
**(from the interactive node: qrsh)**
  - ▶ Sort the .psl file

```
cat siskin_blat.psl | sort -n -k 16,16 | sort -n -k 14,14 > siskin_blat_srt.psl
```
  - ▶ Load the augustus/3.3 module

```
module load bioinformatics/augustus/3.3
```
  - ▶ Run the script blat2hints.pl

```
blat2hints.pl --in=siskin_blat_srt.psl --out=siskin_blat_hints.out
```

# Augustus hints: RepeatMasker

- ▶ cd to your repeatmasker folder
- ▶ load the module repeatmasker  
module load bioinformatics/repeatmasker
- ▶ Use the script rmOutToGFF3.pl to convert your .out file into GFF3:

```
rmOutToGFF3.pl <input.out> > <output>.gff3
```

# Augustus hints: RepeatMasker

- ▶ Copy the gff2hints.pl script  
`cp /pool/genomics/tsuchiyam/gff2hints.pl .`
- ▶ Now let's convert the gff3 into a hints file:  
`perl gff2hints.pl --in=input.gff3 --source=RM --  
out=hints_RM.out`

# Augustus: Combining hints

► Merge both files:

1. from your augustus folder
2. merge both hints files

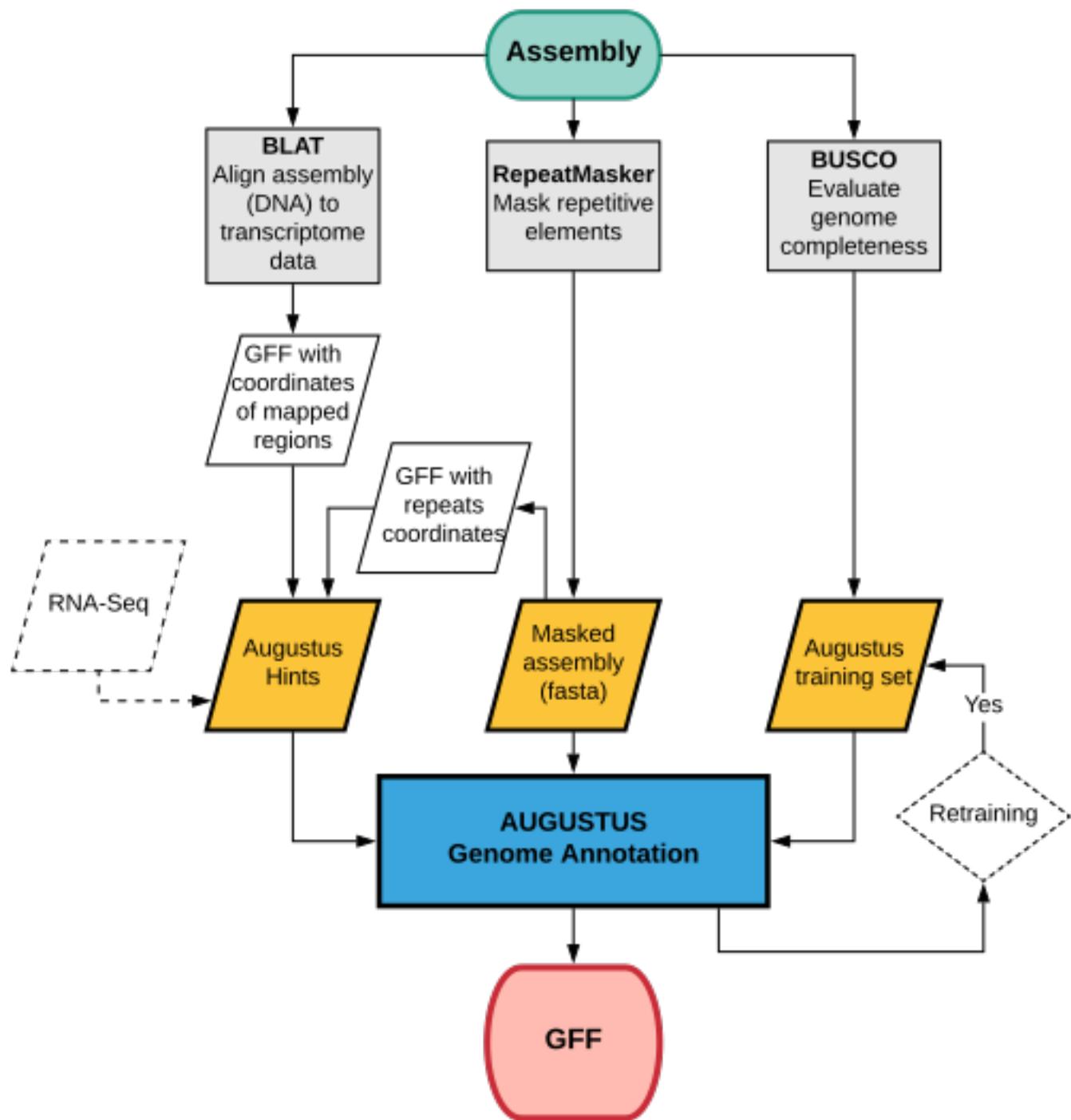
```
cat ../repeatmasker/hints_RM.out  
blat/siskin_blat_hints.out | sort -n -k 4,4 | sort -n  
-k 5,5 > siskin_RM_E.hints
```



# Phew...



- ▶ What do we have now?
  - ▶ Masked fasta from RepeatMasker
  - ▶ Hints file
  - ▶ Training set from BUSCO
  
- ▶ What else do we need?



# Augustus

- *ab initio* (internal) + evidence-driven(external)

AUGUSTUS is based on a generalized hidden Markov model (GHMM), which defines probability distributions for the various sections of genomic sequences. Introns, exons, intergenic regions, etc. correspond to states in the model and each state is thought to create DNA sequences with certain pre-defined emission probabilities.

(Mario Stanke, Burkhard Morgenstern; AUGUSTUS: a web server for gene prediction in eukaryotes that allows user-defined constraints, *Nucleic Acids Research*, Volume 33, Issue suppl\_2, 1 July 2005, Pages W465–W467, <https://doi.org/10.1093/nar/gki458>)

# Augustus

- ▶ Augustus needs to be trained:
  - ▶ Training consists on the generation of a training set that will more accurately predict genes.

(BUSCO solved the training issue for us)

# Augustus

## ► Inputs:

- Masked fasta (repeatmasker output)
- Hints file
- Training set

# Augustus

- ▶ Masked fasta:
  - ▶ Copy the file from the repeatmasker  
`cp .../repeatmasker/assembly.fasta.masked .`

Alternatively, we can create a symlink instead.

`ln -s .../repeatmasker/assembly.fasta.masked .`

# Augustus extrinsic file

- ▶ Defines how the information from the hints will be weighted:
- ▶ It assigns “bonus” and “malus” (penalty) values to each hint used
  - ▶ M: manual annotation
  - ▶ W: RNA-Seq coverage information
  - ▶ E: EST/cDNA database hit
  - ▶ R: retroposed genes
  - ▶ RM: repeat masking

```
[SOURCES]
M RM E

#
# individual_liability: Only unsatisfiable hints are disregarded. By default this flag is not set
# and the whole hint group is disregarded when one hint in it is unsatisfiable.
# 1group1gene: Try to predict a single gene that covers all hints of a given group. This is relevant for
# hint groups with gaps, e.g. when two ESTs, say 5' and 3', from the same clone align nearby.
#
[SOURCE-PARAMETERS]

# feature          bonus          malus      gradelevelcolumns
#                   r+/r-
#
# the gradelevel colums have the following format for each source
# sourcecharacter numscoreclasses boundary ... boundary gradequot ... gradequot
#


[GENERAL]
  start    1        1 M    1 1e+100  RM 1    1   E 1    1
  stop     1        1 M    1 1e+100  RM 1    1   E 1    1
  tss      1        1 M    1 1e+100  RM 1    1   E 1    1
  tts      1        1 M    1 1e+100  RM 1    1   E 1    1
  ass      1        1 0.1 M   1 1e+100  RM 1    1   E 1    1
  dss      1        1 0.1 M   1 1e+100  RM 1    1   E 1    1
  exonpart 1        .992 .985 M   1 1e+100  RM 1    1   E 1    1e2
  exon     1        1 M    1 1e+100  RM 1    1   E 1    1e4
  intronpart 1        1 M    1 1e+100  RM 1    1   E 1    1
  intron   1        .34 M   1 1e+100  RM 1    1   E 1    1e6
  CDSpart  1        1 .985 M   1 1e+100  RM 1    1   E 1    1
  CDS      1        1 M    1 1e+100  RM 1    1   E 1    1
  UTRpart  1        1 .985 M   1 1e+100  RM 1    1   E 1    1
  UTR      1        1 M    1 1e+100  RM 1    1   E 1    1
  irpart   1        1 M    1 1e+100  RM 1    1   E 1    1
  nonexonpart 1        1 M    1 1e+100  RM 1    1.15 E 1    1
  genicpart 1        1 M    1 1e+100  RM 1    1   E 1    1

#
# Explanation: see original extrinsic.cfg file
#
```

# Extrinsic file

- ▶ For practical purposes, copy the extrinsic file below to your augustus/config/extrinsic folder:

/scratch/genomics/tsuchiyam/augustus/config  
/extrinsic/extrinsic.M.RM.E.cfg

# Let's run Augustus

```
augustus --strand=both --singlestrand=true \
--hintsfile= siskin_RM_E.hints \
--extrinsicCfgFile=extrinsic.M.RM.E.cfg \
--alternatives-from-evidence=true \
--gff3=on \
--uniqueGenelId=true \
--softmasking=1 \
--species= BUSCO_siskin_busco_2684346740 \
assembly.fasta.masked > siskin_Contig3141.gff
```



# What do we do with the annotation?

- ▶ The next step is to understand the functions of the genes we identified
- ▶ For functional annotation, we used both BLAST and BLAST2GO

# Galaxy PROJECT

Galaxy

Analyze Data Workflow Visualize Shared Data Help Login or Register

Using 0%

Tools

search tools

[Get Data](#)

[Lift-Over](#)

[Collection Operations](#)

[Text Manipulation](#)

[Datamash](#)

[Convert Formats](#)

[Filter and Sort](#)

[Join, Subtract and Group](#)

[Fetch Alignments / Sequences](#)

[NGS: QC and manipulation](#)

[NGS: DeepTools](#)

[NGS: Mapping](#)

[NGS: RNA Analysis](#)

[NGS: SAMtools](#)

[NGS: BamTools](#)

[NGS: Picard](#)

[NGS: VCF Manipulation](#)

[NGS: Peak Calling](#)

[NGS: Variant Analysis](#)

[NGS: RNA Structure](#)

[NGS: Du Novo](#)

[NGS: Gemini](#)

Galaxy is an open source, web-based platform for data intensive biomedical research. If you are new to Galaxy [start here](#) or consult our help resources. You can install your own Galaxy by following the [tutorial](#) and choose from thousands of tools from the [Tool Shed](#).

Want help?  
Get answers.

 **Biostars**  
GALAXY EXPLAINED

Tweets by [@galaxyproject](#) 6

 Galaxy Project  
@galaxyproject

"Galaxy for NGS Data Analysis" workshop Nov 13-14  
@UCLA from @UCLAQCBio  
[qcb.ucla.edu/collaboratory/](http://qcb.ucla.edu/collaboratory/) ... #usegalaxy

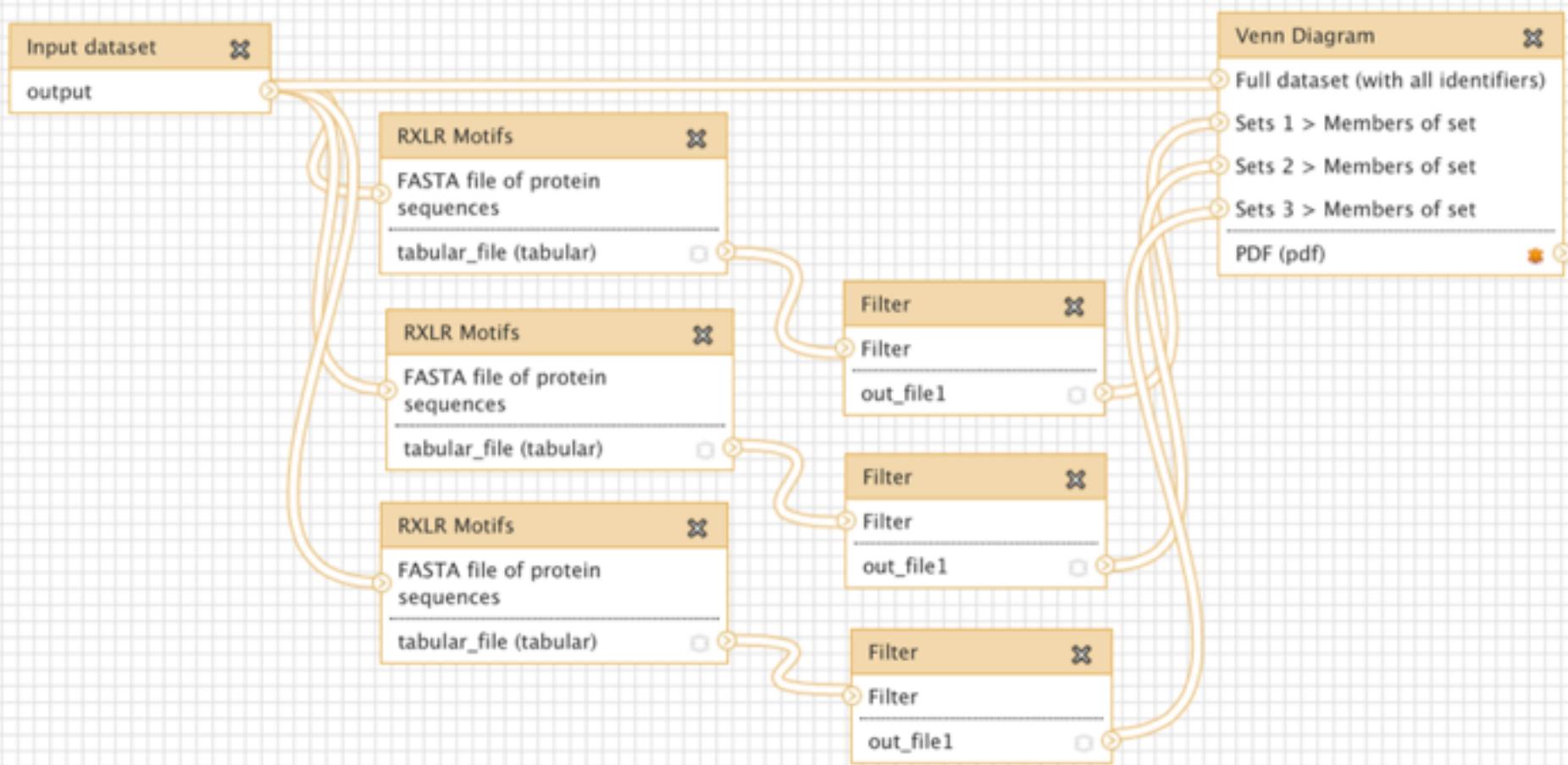
History

search datasets

Unnamed history  
(empty)

This history is empty. You can [load your own data](#) or [get data from an external source](#)

# Galaxy PROJECT

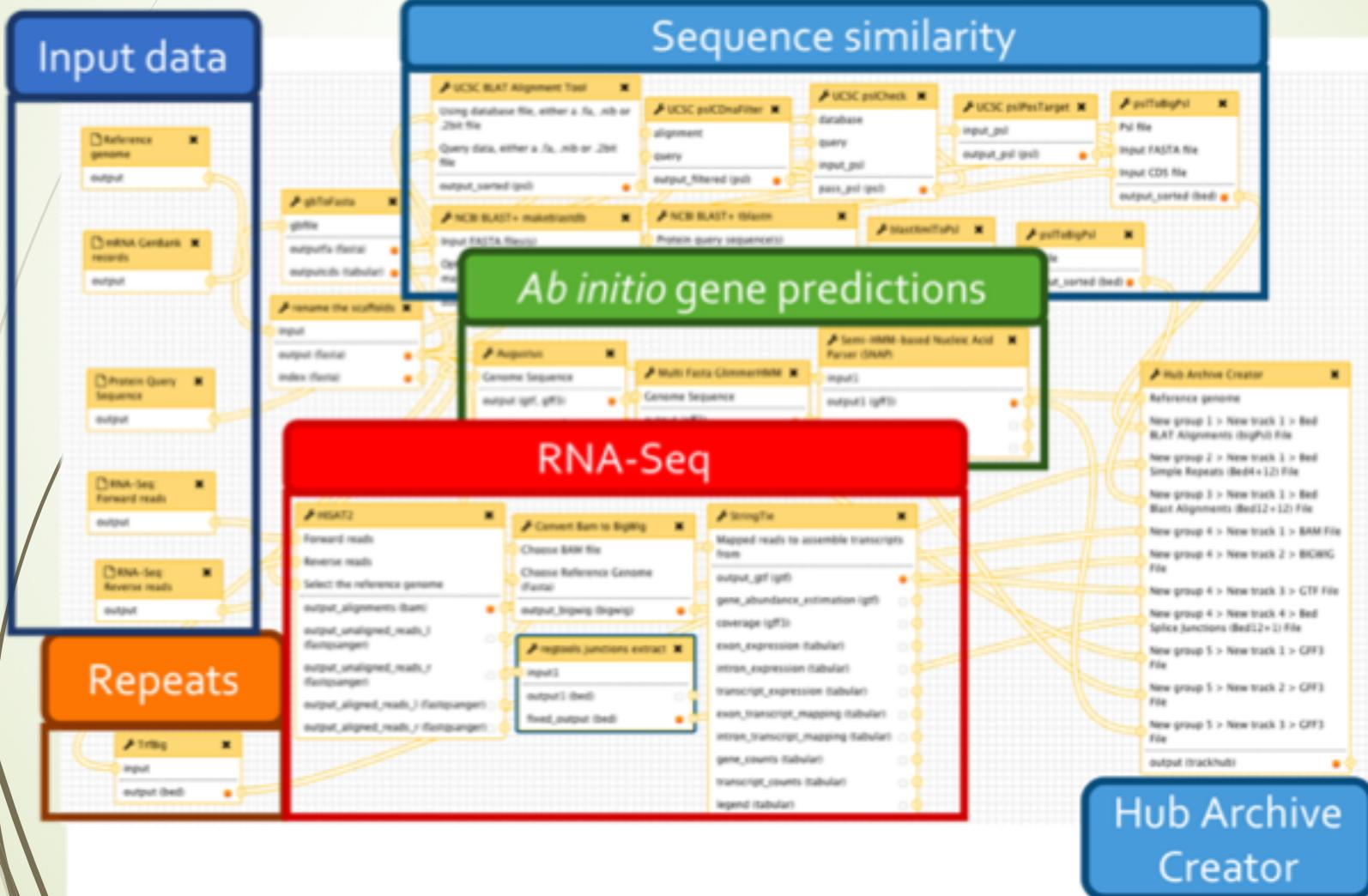




- ▶ Usegalaxy.org
- ▶ Core group + Galaxy community (very active)
- ▶ At Smithsonian:
  - ▶ Galaxy Server (connected to Hydra)
    - ▶ Powerful
    - ▶ Behind firewall, intranet only
  - ▶ Galaxy on the cloud
    - ▶ Accessible from anywhere

# Galaxy PROJECT

## G-OnRamp





**Galaxy Annotation:**  
**<https://usegalaxy.eu/>**

# Thank you



Bug #415



Bug #416



Bug #417



Bug #418



Bug #419



Bug #420