**Genome annotation**

Use Maker for genome annotation

http://www.yandell-lab.org/software/maker.html

http://weatherby.genetics.utah.edu/MAKER/wiki/index.php/Main\_Page

http://www.molecularevolution.org/molevolfiles/exercises/augustus/training.html

### From Tim’s GitHub: <https://github.com/tsackton/ratite-genomics/tree/master/annotation>

##########################################################################

To annotate protein-coding genes in our newly assembled genomes, we used MAKER v2.31.8. We used a 2-step approach where we first annotated the emu and Chilean tinamou genomes, for which we have RNA-seq data, used the initial annotations to build improved SNAP and AUGUSTUS models, and then annotated all 10 species using the improved models. For the species without same-species RNA-seq data, we use the closest appropriate alternate-species RNA-seq as evidence, but we do not retrain gene models for these species.

**Initial MAKER run.**

For the initial MAKER run, we used the Augustus chicken models distributed with Augustus, and generated chicken-trained SNAP models ourselves. See maker/snap/train\_snap\_chicken.sh for details on training.

The control files for the MAKER runs are in maker/run1/.

**Retraining**

We retrained both SNAP and AUGUSTUS based on high-confidence gene models from the initial maker run for notPer and droNov (the two species we have RNA-seq data for). See the maker/snap/ and maker/augustus/ subdirectories for code to do this retraining.

**Final MAKER run.**

For our final MAKER run, we used the trained gene predictors, as well as expanded evidence sources (TopHat junctions in additional to Trinity assemblies). The control files for the final MAKER runs are in maker/run2/.

##########################################################################

### train\_snap\_chicken.sh

##########################################################################

#!/bin/bash

#code to do initial SNAP training on chicken

#1. train SNAP on chicken: (based on https://biowize.wordpress.com/2012/06/01/training-the-snap-ab-initio-gene-predictor/ and SNAP readme)

#a. Get chicken genome, chromosomes only:

for CHR in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 Z

do

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF\_000002315.3\_Gallus\_gallus-4.0/GCF\_000002315.3\_Gallus\_gallus-4.0\_assembly\_structure/Primary\_Assembly/assembled\_chromosomes/FASTA/chr$CHR.fna.gz

done

cat \*.gz > galgal4chr.fa.gz

gunzip galgal4chr.fa.gz

rm \*.gz

#b. Get chicken GFF from NCBI

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF\_000002315.3\_Gallus\_gallus-4.0/GCF\_000002315.3\_Gallus\_gallus-4.0\_genomic.gff.gz

gunzip GCF\_000002315.3\_Gallus\_gallus-4.0\_genomic.gff.gz

#c. Filter chicken GFF

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF\_000002315.3\_Gallus\_gallus-4.0/GCF\_000002315.3\_Gallus\_gallus-4.0\_assembly\_structure/Primary\_Assembly/assembled\_chromosomes/chr2acc

for ACC in $(egrep "^[0-9Z]" chr2acc | grep -v "^32" | cut -f2,2)

do

egrep "^$ACC" GCF\_000002315.3\_Gallus\_gallus-4.0\_genomic.gff >> galgal4\_chr.gff

done

grep "BestRefSeq" galgal4\_chr.gff > galgal4\_filt\_chr.gff

grep -v "exception=" galgal4\_filt\_chr.gff > galgal4\_filt\_chr\_good.gff

#use gffread to remove cds with stop codons and other problems

module load cufflinks

gffread galgal4\_filt\_chr\_good.gff -g galgal4chr.fa -o galgal4\_final.gff -V -C

#merge fasta info to end of gff

echo "##FASTA" >> galgal4\_final.gff

cat galgal4chr.fa >> galgal4\_final.gff

#d. convert gff to zff

maker2zff -n galgal4\_final.gff

#e. check quality of zff files

fathom genome.ann genome.dna -gene-stats > gene-stats.log 2>&1

#MODEL589 skipped due to errors

#MODEL1127 1 1 4 - errors(2): gene:misordered\_Einit gene:misordered\_Eterm

#MODEL1127 skipped due to errors

#MODEL2463 1 1 12 - errors(2): gene:misordered\_Einit gene:misordered\_Eterm

#MODEL2463 skipped due to errors

fathom genome.ann genome.dna -validate > validate.log 2>&1

grep -v "MODEL589" genome.ann | grep -v "MODEL1127" | grep -v "MODEL2463" > genome.fixed.ann

mv genome.ann genome.ann.old

mv genome.fixed.ann genome.ann

fathom genome.ann genome.dna -categorize 1000 > categorize.log 2>&1

fathom uni.ann uni.dna -export 1000 -plus > uni-plus.log 2>&1

#f. run SNAP training algo

mkdir params

cd params

forge ../export.ann ../export.dna > ../forge.log 2>&1

cd ..

hmm-assembler.pl chicken params/ > chicken.hmm

##########################################################################

### Tim had done this already so copy from his ratite folder

cp /n/holylfs/LABS/edwards\_lab/Users/tsackton/ratites/annotation/training/chicken/chicken.hmm /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

### use below can download the chicken annotation folder from ncbi

wget --recursive ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/002/315/GCA\_000002315.3\_Gallus\_gallus-5.0/GCA\_000002315.3\_Gallus\_gallus-5.0\_assembly\_structure

### can download individual chromosome use this:

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/002/315/GCA\_000002315.3\_Gallus\_gallus-5.0/GCA\_000002315.3\_Gallus\_gallus-5.0\_assembly\_structure/Primary\_Assembly/assembled\_chromosomes/FASTA/chrW.fna.gz

cd /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

maker -CTL

This creates three files (type ls -1 to see).

maker\_exe.ctl - contains the path information for the underlying executables.

maker\_bopt.ctl - contains filtering statistics for BLAST and Exonerate

maker\_opt.ctl - contains all other information for MAKER, including the location of the input genome file.

cat maker\_exe.ctl

##########################################################################

#-----Location of Executables Used by MAKER/EVALUATOR

makeblastdb=/n/sw/fasrcsw/apps/Core/ncbi-blast/2.2.30+-fasrc01/bin/makeblastdb #location of NCBI+ makeblastdb executable

blastn=/n/sw/fasrcsw/apps/Core/ncbi-blast/2.2.30+-fasrc01/bin/blastn #location of NCBI+ blastn executable

blastx=/n/sw/fasrcsw/apps/Core/ncbi-blast/2.2.30+-fasrc01/bin/blastx #location of NCBI+ blastx executable

tblastx=/n/sw/fasrcsw/apps/Core/ncbi-blast/2.2.30+-fasrc01/bin/tblastx #location of NCBI+ tblastx executable

formatdb= #location of NCBI formatdb executable

blastall= #location of NCBI blastall executable

xdformat= #location of WUBLAST xdformat executable

blasta= #location of WUBLAST blasta executable

RepeatMasker=/n/sw/fasrcsw/apps/Core/RepeatMasker/4.0.5-fasrc03/RepeatMasker #location of RepeatMasker executable

exonerate=/n/sw/fasrcsw/apps/Core/exonerate/2.4.0-fasrc01/bin/exonerate #location of exonerate executable

#-----Ab-initio Gene Prediction Algorithms

snap=/n/sw/fasrcsw/apps/Core/snap/2013.11.29-fasrc01/snap #location of snap executable

gmhmme3= #location of eukaryotic genemark executable

gmhmmp= #location of prokaryotic genemark executable

augustus=/n/sw/fasrcsw/apps/Core/augustus/3.0.3-fasrc02/bin/augustus #location of augustus executable

fgenesh= #location of fgenesh executable

tRNAscan-SE= #location of trnascan executable

snoscan= #location of snoscan executable

#-----Other Algorithms

probuild= #location of probuild executable (required for genemark)

##########################################################################

cat maker\_bopts.ctl

##########################################################################

#-----BLAST and Exonerate Statistics Thresholds

blast\_type=ncbi+ #set to 'ncbi+', 'ncbi' or 'wublast'

pcov\_blastn=0.8 #Blastn Percent Coverage Threhold EST-Genome Alignments

pid\_blastn=0.85 #Blastn Percent Identity Threshold EST-Genome Aligments

eval\_blastn=1e-10 #Blastn eval cutoff

bit\_blastn=40 #Blastn bit cutoff

depth\_blastn=0 #Blastn depth cutoff (0 to disable cutoff)

pcov\_blastx=0.5 #Blastx Percent Coverage Threhold Protein-Genome Alignments

pid\_blastx=0.4 #Blastx Percent Identity Threshold Protein-Genome Aligments

eval\_blastx=1e-06 #Blastx eval cutoff

bit\_blastx=30 #Blastx bit cutoff

depth\_blastx=0 #Blastx depth cutoff (0 to disable cutoff)

pcov\_tblastx=0.8 #tBlastx Percent Coverage Threhold alt-EST-Genome Alignments

pid\_tblastx=0.85 #tBlastx Percent Identity Threshold alt-EST-Genome Aligments

eval\_tblastx=1e-10 #tBlastx eval cutoff

bit\_tblastx=40 #tBlastx bit cutoff

depth\_tblastx=0 #tBlastx depth cutoff (0 to disable cutoff)

pcov\_rm\_blastx=0.5 #Blastx Percent Coverage Threhold For Transposable Element Masking

pid\_rm\_blastx=0.4 #Blastx Percent Identity Threshold For Transposbale Element Masking

eval\_rm\_blastx=1e-06 #Blastx eval cutoff for transposable element masking

bit\_rm\_blastx=30 #Blastx bit cutoff for transposable element masking

ep\_score\_limit=20 #Exonerate protein percent of maximal score threshold

en\_score\_limit=20 #Exonerate nucleotide percent of maximal score threshold

##########################################################################

cp /n/holylfs/LABS/edwards\_lab/Users/ywsin/petrel\_assembly\_from\_Tim/final.assembly.fasta /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

mv final.assembly.fasta LESP\_genome.fasta

cp /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/trinity\_output2/Trinity.fasta /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

mv Trinity.fasta LESP\_transcriptome.fasta

# For the TopHat junction file

#process junctions.bed to keep only high-quality junctions (score > 5) and then convert to gffs

cp /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/tophat\_out1/junctions.bed /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/tophat\_junction\_file

# template from Tim

####################

#first filter with awk

awk '{if($5 >= 5) print $0}' $BED > $TARGET.junctions.bed

#convert to GFF

tophat2gff3 $BED > $BED.gff

####################

#first filter with awk

awk '{if($5 >= 5) print $0}' junctions.bed > /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/tophat\_junction\_file/high\_quality\_junctions.bed

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

#convert to GFF

tophat2gff3 high\_quality\_junctions.bed > high\_quality\_tophat\_junctions.gff

cp /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/tophat\_junction\_file/high\_quality\_tophat\_junctions.gff /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

# Copy the repeat protein files

cp /n/home12/tsackton/sw/progs/maker/data/te\_proteins.fasta /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

cd /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/protein\_multiple\_organisms

### Download protein list from different species

wget ftp://ftp.ncbi.nlm.nih.gov/refseq/release/complete/\*.faa.gz

# Concatenate the protein fasta files

"find ./ -name \\*.gz -exec gunzip -k {} \;"

"cat \*.faa > ~/output.txt"

gunzip \*.gz

cat \*.protein.faa > all\_protein.faa

# This is all protein seq on ncbi which is too much, just download some good quality genome proteins

# Gallus gallus (chicken)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/002/315/GCF\_000002315.4\_Gallus\_gallus-5.0/GCF\_000002315.4\_Gallus\_gallus-5.0\_protein.faa.gz

Scaffold N50 6,379,610

Contig N50 2,894,815

gunzip GCF\_000002315.4\_Gallus\_gallus-5.0\_protein.faa.gz

# Taeniopygia guttata (zebra finch)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/151/805/GCF\_000151805.1\_Taeniopygia\_guttata-3.2.4/GCF\_000151805.1\_Taeniopygia\_guttata-3.2.4\_protein.faa.gz

Scaffold N50 8,236,790

Contig N50 38,639

gunzip GCF\_000151805.1\_Taeniopygia\_guttata-3.2.4\_protein.faa.gz

# Ficedula albicollis (collared flycatcher)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/247/815/GCA\_000247815.2\_FicAlb1.5/GCA\_000247815.2\_FicAlb1.5\_protein.faa.gz

Scaffold N50 5,639

Contig N50 4,718

# Pseudopodoces humilis (Tibetan ground-tit)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/331/425/GCF\_000331425.1\_PseHum1.0/GCF\_000331425.1\_PseHum1.0\_protein.faa.gz

Scaffold N50 16,337,386

Contig N50 165,265

gunzip GCF\_000331425.1\_PseHum1.0\_protein.faa.gz

# Aquila chrysaetos (golden eagle)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/766/835/GCF\_000766835.1\_Aquila\_chrysaetos-1.0.2/GCF\_000766835.1\_Aquila\_chrysaetos-1.0.2\_protein.faa.gz

Scaffold N50 9,230,743

Contig N50 172,329

gunzip GCF\_000766835.1\_Aquila\_chrysaetos-1.0.2\_protein.faa.gz

# Melopsittacus undulatus (budgerigar)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/238/935/GCF\_000238935.1\_Melopsittacus\_undulatus\_6.3/GCF\_000238935.1\_Melopsittacus\_undulatus\_6.3\_protein.faa.gz

Scaffold N50 10,614,383

Contig N50 55,633

gunzip GCF\_000238935.1\_Melopsittacus\_undulatus\_6.3\_protein.faa.gz

# Aptenodytes forsteri (emperor penguin)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/699/145/GCF\_000699145.1\_ASM69914v1/GCF\_000699145.1\_ASM69914v1\_protein.faa.gz

Scaffold N50 5,071,598

Contig N50 31,730

gunzip GCF\_000699145.1\_ASM69914v1\_protein.faa.gz

# Pygoscelis adeliae (Adelie penguin)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/699/105/GCF\_000699105.1\_ASM69910v1/GCF\_000699105.1\_ASM69910v1\_protein.faa.gz

Scaffold N50 5,118,896

Contig N50 22,195

# Cathartes aura (turkey vulture)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/699/945/GCA\_000699945.1\_ASM69994v1/GCA\_000699945.1\_ASM69994v1\_protein.faa.gz

Scaffold N50 36,359

Contig N50 15,248

# Pelecanus crispus (Dalmatian pelican)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/687/375/GCF\_000687375.1\_ASM68737v1/GCF\_000687375.1\_ASM68737v1\_protein.faa.gz

Scaffold N50 43,364

Contig N50 21,679

# Phalacrocorax carbo (great cormorant)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/708/925/GCF\_000708925.1\_ASM70892v1/GCF\_000708925.1\_ASM70892v1\_protein.faa.gz

Scaffold N50 48,427

Contig N50 17,343

# Fulmarus glacialis (northern fulmar)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/690/835/GCF\_000690835.1\_ASM69083v1/GCF\_000690835.1\_ASM69083v1\_protein.faa.gz

Scaffold N50 47,208

Contig N50 25,926

# Gavia stellata (red-throated loon)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/690/875/GCF\_000690875.1\_ASM69087v1/GCF\_000690875.1\_ASM69087v1\_protein.faa.gz

Scaffold N50 45,523

Contig N50 24,321

# Phaethon lepturus (white-tailed tropicbird)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/687/285/GCF\_000687285.1\_ASM68728v1/GCF\_000687285.1\_ASM68728v1\_protein.faa.gz

Scaffold N50 47,896

Contig N50 22,941

# Charadrius vociferus (killdeer)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/708/025/GCF\_000708025.1\_ASM70802v2/GCF\_000708025.1\_ASM70802v2\_protein.faa.gz

Scaffold N50 3,657,050

Contig N50 39,278

gunzip GCF\_000708025.1\_ASM70802v2\_protein.faa.gz

#Podiceps cristatus (great crested grebe)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/699/545/GCA\_000699545.1\_ASM69954v1/GCA\_000699545.1\_ASM69954v1\_protein.faa.gz

Scaffold N50 30,087

Contig N50 17,412

#Phoenicopterus ruber (American flamingo)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/687/265/GCA\_000687265.1\_ASM68726v1/GCA\_000687265.1\_ASM68726v1\_protein.faa.gz

Scaffold N50 38,071

Contig N50 20,262

#Anas platyrhynchos (mallard)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/355/885/GCF\_000355885.1\_BGI\_duck\_1.0/GCF\_000355885.1\_BGI\_duck\_1.0\_protein.faa.gz

Scaffold N50 1,233,631

Contig N50 26,114

gunzip GCF\_000355885.1\_BGI\_duck\_1.0\_protein.faa.gz

#Struthio camelus (African ostrich)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/698/965/GCF\_000698965.1\_ASM69896v1/GCF\_000698965.1\_ASM69896v1\_protein.faa.gz

Scaffold N50 3,593,425

Contig N50 34,997

#Serinus canaria (common canary)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/534/875/GCF\_000534875.1\_SCA1/GCF\_000534875.1\_SCA1\_protein.faa.gz

Scaffold N50 17,815,079

Contig N50 53,884

#Zonotrichia albicollis (white-throated sparrow)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/385/455/GCF\_000385455.1\_Zonotrichia\_albicollis-1.0.1/GCF\_000385455.1\_Zonotrichia\_albicollis-1.0.1\_protein.faa.gz

Scaffold N50 4,866,725

Contig N50 112,748

gunzip GCF\_000385455.1\_Zonotrichia\_albicollis-1.0.1\_protein.faa.gz

#Corvus cornix (hooded crow)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/738/735/GCF\_000738735.1\_Hooded\_Crow\_genome/GCF\_000738735.1\_Hooded\_Crow\_genome\_protein.faa.gz

Scaffold N50 16,358,221

Contig N50 94,375

#Nipponia nippon (crested ibis)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/708/225/GCF\_000708225.1\_ASM70822v1/GCF\_000708225.1\_ASM70822v1\_protein.faa.gz

Scaffold N50 5,211,696

Contig N50 29,116

gunzip GCF\_000708225.1\_ASM70822v1\_protein.faa.gz

#Anolis carolinensis (green anole)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/090/745/GCF\_000090745.1\_AnoCar2.0/GCF\_000090745.1\_AnoCar2.0\_protein.faa.gz

Scaffold N50 4,033,265

Contig N50 79,867

gunzip GCF\_000090745.1\_AnoCar2.0\_protein.faa.gz

#Crocodylus porosus (Australian saltwater crocodile)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/001/723/895/GCF\_001723895.1\_CroPor\_comp1/GCF\_001723895.1\_CroPor\_comp1\_protein.faa.gz

Scaffold N50 84,437,661

Contig N50 34,073

gunzip GCF\_001723895.1\_CroPor\_comp1\_protein.faa.gz

# Malaclemys terrapin (diamondback terrapin)

ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/001/728/815/GCA\_001728815.2\_terp\_v2\_2/GCA\_001728815.2\_terp\_v2\_2\_protein.faa.gz

Contig N50 437,266

#Chrysemys picta (painted turtle)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/241/765/GCF\_000241765.3\_Chrysemys\_picta\_bellii-3.0.3/GCF\_000241765.3\_Chrysemys\_picta\_bellii-3.0.3\_protein.faa.gz

Scaffold N50 6,605,846

Contig N50 21,349

gunzip GCF\_000241765.3\_Chrysemys\_picta\_bellii-3.0.3\_protein.faa.gz

#Homo sapiens (human)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/001/405/GCF\_000001405.35\_GRCh38.p9/GCF\_000001405.35\_GRCh38.p9\_protein.faa.gz

Scaffold N50 59,364,414

Contig N50 56,413,054

gunzip GCF\_000001405.35\_GRCh38.p9\_protein.faa.gz

#Mus musculus (house mouse)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/001/635/GCF\_000001635.25\_GRCm38.p5/GCF\_000001635.25\_GRCm38.p5\_protein.faa.gz

Scaffold N50 52,589,046

Contig N50 32,273,079

gunzip GCF\_000001635.25\_GRCm38.p5\_protein.faa.gz

#Danio rerio (zebrafish)

wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/002/035/GCF\_000002035.5\_GRCz10/GCF\_000002035.5\_GRCz10\_protein.faa.gz

Scaffold N50 2,181,225

Contig N50 1,258,148

gunzip GCF\_000002035.5\_GRCz10\_protein.faa.gz

cat \*protein.faa > all\_protein\_multiple\_organisms.faa

cp /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/protein\_multiple\_organisms/all\_protein\_multiple\_organisms.faa /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

cat maker\_opts.ctl

##########################################################################

#-----Genome (these are always required)

genome=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.fasta #genome sequence (fasta file or fasta embeded in GFF3 file)

organism\_type=eukaryotic #eukaryotic or prokaryotic. Default is eukaryotic

#-----Re-annotation Using MAKER Derived GFF3

maker\_gff= #MAKER derived GFF3 file

est\_pass=0 #use ESTs in maker\_gff: 1 = yes, 0 = no

altest\_pass=0 #use alternate organism ESTs in maker\_gff: 1 = yes, 0 = no

protein\_pass=0 #use protein alignments in maker\_gff: 1 = yes, 0 = no

rm\_pass=0 #use repeats in maker\_gff: 1 = yes, 0 = no

model\_pass=0 #use gene models in maker\_gff: 1 = yes, 0 = no

pred\_pass=0 #use ab-initio predictions in maker\_gff: 1 = yes, 0 = no

other\_pass=0 #passthrough anyything else in maker\_gff: 1 = yes, 0 = no

#-----EST Evidence (for best results provide a file for at least one)

est=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_transcriptome.fasta #set of ESTs or assembled mRNA-seq in fasta format

altest= #EST/cDNA sequence file in fasta format from an alternate organism

est\_gff=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/high\_quality\_tophat\_junctions.gff #aligned ESTs or mRNA-seq from an external GFF3 file (TopHat junction file goes to here)

altest\_gff= #aligned ESTs from a closly relate species in GFF3 format

#-----Protein Homology Evidence (for best results provide a file for at least one)

protein=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/all\_protein\_multiple\_organisms.faa #protein sequence file in fasta format (i.e. from mutiple oransisms)

protein\_gff= #aligned protein homology evidence from an external GFF3 file

#-----Repeat Masking (leave values blank to skip repeat masking)

model\_org=aves #select a model organism for RepBase masking in RepeatMasker (replace all with aves)

rmlib= #provide an organism specific repeat library in fasta format for RepeatMasker

repeat\_protein=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/te\_proteins.fasta #provide a fasta file of transposable element proteins for RepeatRunner

rm\_gff= #pre-identified repeat elements from an external GFF3 file

prok\_rm=0 #forces MAKER to repeatmask prokaryotes (no reason to change this), 1 = yes, 0 = no

softmask=1 #use soft-masking rather than hard-masking in BLAST (i.e. seg and dust filtering)

#-----Gene Prediction

snaphmm=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/chicken.hmm #SNAP HMM file

gmhmm= #GeneMark HMM file

augustus\_species=chicken #Augustus gene prediction species model

fgenesh\_par\_file= #FGENESH parameter file

pred\_gff= #ab-initio predictions from an external GFF3 file

model\_gff= #annotated gene models from an external GFF3 file (annotation pass-through)

est2genome=0 #infer gene predictions directly from ESTs, 1 = yes, 0 = no

protein2genome=0 #infer predictions from protein homology, 1 = yes, 0 = no

trna=0 #find tRNAs with tRNAscan, 1 = yes, 0 = no

snoscan\_rrna= #rRNA file to have Snoscan find snoRNAs

unmask=0 #also run ab-initio prediction programs on unmasked sequence, 1 = yes, 0 = no

#-----Other Annotation Feature Types (features MAKER doesn't recognize)

other\_gff= #extra features to pass-through to final MAKER generated GFF3 file

#-----External Application Behavior Options

alt\_peptide=C #amino acid used to replace non-standard amino acids in BLAST databases

cpus=1 #max number of cpus to use in BLAST and RepeatMasker (not for MPI, leave 1 when using MPI)

#-----MAKER Behavior Options

max\_dna\_len=100000 #length for dividing up contigs into chunks (increases/decreases memory usage)(Tim used 300000)

min\_contig=1 #skip genome contigs below this length (under 10kb are often useless)(Tim used 1000 because AllPaths minimum contig size is 1000 so actually no difference for 1 and 1000)

pred\_flank=200 #flank for extending evidence clusters sent to gene predictors

pred\_stats=0 #report AED and QI statistics for all predictions as well as models

AED\_threshold=1 #Maximum Annotation Edit Distance allowed (bound by 0 and 1)

min\_protein=0 #require at least this many amino acids in predicted proteins (Tim used 20)

alt\_splice=1 #Take extra steps to try and find alternative splicing, 1 = yes, 0 = no (change 0 to 1)

always\_complete=0 #extra steps to force start and stop codons, 1 = yes, 0 = no

map\_forward=0 #map names and attributes forward from old GFF3 genes, 1 = yes, 0 = no

keep\_preds=0 #Concordance threshold to add unsupported gene prediction (bound by 0 and 1)

split\_hit=10000 #length for the splitting of hits (expected max intron size for evidence alignments)

single\_exon=0 #consider single exon EST evidence when generating annotations, 1 = yes, 0 = no

single\_length=250 #min length required for single exon ESTs if 'single\_exon is enabled'

correct\_est\_fusion=0 #limits use of ESTs in annotation to avoid fusion genes

tries=5 #number of times to try a contig if there is a failure for some reason (change 2 to 5)

clean\_try=0 #remove all data from previous run before retrying, 1 = yes, 0 = no

clean\_up=0 #removes theVoid directory with individual analysis files, 1 = yes, 0 = no

TMP= #specify a directory other than the system default temporary directory for temporary files

##########################################################################

# All files in this directory

cd /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker

# Test Maker really works using interactive first

srun -p interact --pty --mem 10000 -t 0-06:00 /bin/bash

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

maker -fix\_nucleotides

### Tim’s mpi script

### maker\_mpi.sh

##########################################################################

#!/bin/bash

#SBATCH -p general

#SBATCH --mem-per-cpu 6000

#SBATCH -n 256

#SBATCH -t 6-00:00

#SBATCH -o maker\_%A.out

#SBATCH -e maker\_%A.err

#SBATCH -J mkerMPI

#set up and log species

SP=$1

echo "Running MAKER on $SP."

#load modules

source new-modules.sh

#module load openmpi/1.8.3-fasrc02

module load gcc/6.2.0-fasrc01 openmpi/2.0.1-fasrc01

export LD\_PRELOAD=/n/sw/fasrcsw/apps/Comp/gcc/4.8.2-fasrc01/openmpi/1.8.3-fasrc02/lib64/libmpi.so

#change to correct directory

cd /n/regal/edwards\_lab/ratites/maker2/annotation/$SP

#run MAKER

mpiexec -mca btl ^openib -np 256 maker maker\_opts\_$SP.ctl maker\_bopts.ctl maker\_exe.ctl -fix\_nucleotides

##########################################################################

### Or

### apthaa\_maker\_mpi.sh

##########################################################################

#!/bin/bash

#SBATCH -p general

#SBATCH --mem-per-cpu 8000

#SBATCH -n 128

#SBATCH -t 5-00:00

#SBATCH -o maker\_mpi3.out

#SBATCH -e maker\_mpi3.err

#SBATCH -J makerMPI

source new-modules.sh

#module load openmpi/1.8.3-fasrc02

module load gcc/6.2.0-fasrc01 openmpi/2.0.1-fasrc01

~~export LD\_PRELOAD=/n/sw/fasrcsw/apps/Comp/gcc/4.8.2-fasrc01/openmpi/1.8.3-fasrc02/lib64/libmpi.so~~

mpiexec ~~-mca btl ^openib~~ -np 128 maker -fix\_nucleotides

##########################################################################

### Test mpi works first by requestion a small amount of resources

### maker\_LESP\_mpi\_test.sh

##########################################################################

#!/bin/bash

#SBATCH -p general

#SBATCH --mem-per-cpu 6000

#SBATCH -n 2

#SBATCH -t 02:00:00

#SBATCH --contiguous # Ensure that all of the cores are on the same Infiniband network

#SBATCH -o maker\_mpi\_test\_%A.out

#SBATCH -e maker\_mpi\_test\_%A.err

#SBATCH -J makerMPItest

#SBATCH --mail-type=ALL # Type of email notification- BEGIN,END,FAIL,ALL

#SBATCH --mail-user=yungwasin@fas.harvard.edu # Email to which notifications will be sent

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

export LD\_PRELOAD=/n/sw/fasrcsw/apps/Comp/gcc/5.2.0-fasrc02/openmpi/2.0.1-fasrc01/lib64/libmpi.so

mpiexec -mca btl ^openib -np 2 maker -fix\_nucleotides

##########################################################################

### maker\_LESP\_mpi1.sh

##########################################################################

#!/bin/bash

#SBATCH -p general

#SBATCH --mem-per-cpu 6000

#SBATCH -n 256

#SBATCH -t 7-00:00:00

#SBATCH --contiguous # Ensure that all of the cores are on the same Infiniband network

#SBATCH -o maker\_LESP\_mpi1\_%A.out

#SBATCH -e maker\_LESP\_mpi1\_%A.err

#SBATCH -J makerLESPMPI1

#SBATCH --mail-type=ALL # Type of email notification- BEGIN,END,FAIL,ALL

#SBATCH --mail-user=yungwasin@fas.harvard.edu # Email to which notifications will be sent

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

export LD\_PRELOAD=/n/sw/fasrcsw/apps/Comp/gcc/5.2.0-fasrc02/openmpi/2.0.1-fasrc01/lib64/libmpi.so

mpiexec -mca btl ^openib -np 256 maker -fix\_nucleotides

##########################################################################

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

# Check the MAKER output

cd /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output

grep 'FINISHED' LESP\_genome\_master\_datastore\_index.log | wc -l

(this will give you the # of scaffolds that maker lists as being 'FINISHED')

> 1697

to double-check the number of input scaffolds (but you already know this):

grep '>' LESP\_genome.fasta | wc -l

to double-check that there's actually output for each scaffold (do this in an interactive session, not on login node):

cd LESP\_genome.maker.output/

find . -type f -name "scaffold\*gff" -printf '%p\n' | xargs ls -lSh > scaffold\_check

This will output file called 'scaffold\_check' that lists all of the scaffold gffs output by maker, with their file size in human readable format, and they will be listed in decreasing order by file size.

So, then, you can do

wc -l scaffold\_check

to double-check that you have the expected 1697 scaffolds (you do), and

tail scaffold\_check

to look at the smallest output file...it has nonzero size, meaning that maker output 'real' (non-blank) files for all of your scaffolds.

### Merge outputs

### Merge files containing all of your output (i.e. a single GFF3 and FASTA file containing all genes)

cd /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output

srun -p interact --pty --mem 20000 -t 0-12:00 /bin/bash

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

~~#fasta\_merge -d dpp\_contig\_master\_datastore\_index.log~~

~~#gff3\_merge -d dpp\_contig\_master\_datastore\_index.log~~

~~#fasta\_merge -d ${SP}.maker.output/${SP}\_master\_datastore\_index.log -o ${SP}.genome~~

~~#gff3\_merge -d ${SP}.maker.output/${SP}\_master\_datastore\_index.log -o ${SP}.genome.gff~~

~~#fasta\_merge -d LESP\_genome.maker.output/LESP\_genome\_master\_datastore\_index.log -o LESP\_genome.genome~~

~~#gff3\_merge -d LESP\_genome.maker.output/LESP\_genome\_master\_datastore\_index.log -o LESP\_genome.genome.gff~~

fasta\_merge -d LESP\_genome\_master\_datastore\_index.log

gff3\_merge -d LESP\_genome\_master\_datastore\_index.log

ls -lh

total 6.0G

-rw-r--r-- 1 ysin edwards\_lab 5.7G Mar 8 01:19 LESP\_genome.all.gff

-rw-r--r-- 1 ysin edwards\_lab 10M Mar 8 01:01 LESP\_genome.all.maker.augustus\_masked.proteins.fasta

-rw-r--r-- 1 ysin edwards\_lab 29M Mar 8 00:59 LESP\_genome.all.maker.augustus\_masked.transcripts.fasta

-rw-r--r-- 1 ysin edwards\_lab 11M Mar 8 01:05 LESP\_genome.all.maker.non\_overlapping\_ab\_initio.proteins.fasta

-rw-r--r-- 1 ysin edwards\_lab 23M Mar 8 01:04 LESP\_genome.all.maker.non\_overlapping\_ab\_initio.transcripts.fasta

-rw-r--r-- 1 ysin edwards\_lab 17M Mar 8 00:58 LESP\_genome.all.maker.proteins.fasta

-rw-r--r-- 1 ysin edwards\_lab 19M Mar 8 01:03 LESP\_genome.all.maker.snap\_masked.proteins.fasta

-rw-r--r-- 1 ysin edwards\_lab 48M Mar 8 01:02 LESP\_genome.all.maker.snap\_masked.transcripts.fasta

-rw-r--r-- 1 ysin edwards\_lab 63M Mar 8 00:57 LESP\_genome.all.maker.transcripts.fasta

drwxr-xr-x 257 ysin edwards\_lab 12K Jan 28 19:10 LESP\_genome\_datastore

-rw-r--r-- 1 ysin edwards\_lab 136M Jan 28 06:19 LESP\_genome.db

-rw-r--r-- 1 ysin edwards\_lab 236K Feb 16 17:01 LESP\_genome\_master\_datastore\_index.log

-rw-r--r-- 1 ysin edwards\_lab 1.4K Feb 16 17:01 maker\_bopts.log

-rw-r--r-- 1 ysin edwards\_lab 1.5K Feb 16 17:01 maker\_exe.log

-rw-r--r-- 1 ysin edwards\_lab 5.1K Feb 16 17:01 maker\_opts.log

drwxr-xr-x 6 ysin edwards\_lab 4.0K Jan 28 06:16 mpi\_blastdb

-rw-r--r-- 1 ysin edwards\_lab 181K Mar 5 02:27 scaffold\_check

-rw-r--r-- 1 ysin edwards\_lab 0 Jan 28 06:19 seen.dbm

mkdir trainsnap

cd trainsnap

~~# gff3\_merge -d ../pyu\_contig1.maker.output/pyu\_contig1\_master\_datastore\_index.log~~

cp /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/LESP\_genome.all.gff /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/trainsnap

# Convert gff to zff

maker2zff LESP\_genome.all.gff

# Check quality of zff files

fathom genome.ann genome.dna -gene-stats > gene-stats.log 2>&1

cat gene-stats.log

412 sequences

0.494864 avg GC fraction (min=0.340039 max=0.743012)

15491 genes (plus=7803 minus=7688)

64 (0.004131) single-exon

15427 (0.995869) multi-exon

148.460236 mean exon (min=1 max=11250)

1878.888062 mean intron (min=4 max=402798)

fathom genome.ann genome.dna -validate > validate.log 2>&1

fathom genome.ann genome.dna -categorize 1000 > categorize.log 2>&1

fathom uni.ann uni.dna -export 1000 -plus > uni-plus.log 2>&1

#fathom -categorize 1000 genome.ann genome.dna

#fathom -export 1000 -plus uni.ann uni.dna

#forge ../export.ann ../export.dna > ../forge.log 2>&1

forge export.ann export.dna > forge.log 2>&1

#forge export.ann export.dna

#hmm-assembler.pl LESP1 trainsnap/ > LESP1.hmm

#hmm-assembler.pl pyu . > ../pyu1.hmm

hmm-assembler.pl LESP . > LESP1.hmm

cd ..

srun -p interact --pty --mem 20000 -t 1-00:00 /bin/bash

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

**##Training Augustus##** (based on: http://www.molecularevolution.org/molevolfiles/exercises/augustus/training.html)

#make GFFs

##NOTE: ../droNov/training\_droNov.ann and similar are the files produced for SNAP training##

##path needs to be set properly for this to work##

cp /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/trainsnap/genome\* /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus

#zff2gff3.pl ../droNov/training\_droNov.ann | perl -plne 's/\t(\S+)$/\t\.\t$1/' > droNov.gff3

zff2gff3.pl /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/genome.ann | perl -plne 's/\t(\S+)$/\t\.\t$1/' > LESP.gff3

#setup

#export AUGUSTUS\_CONFIG\_PATH=/n/sw/fasrcsw/apps/Core/augustus/3.0.3-fasrc02/config

# Set PATH

cp -R /n/sw/fasrcsw/apps/Core/augustus/3.0.3-fasrc02/config /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus

export AUGUSTUS\_CONFIG\_PATH=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/config

new\_species.pl --species=petrel

###Will create parameters for a EUKARYOTIC species!

creating directory /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/config/species/petrel/ ...

creating /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/config/species/petrel/petrel\_parameters.cfg ...

creating /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/config/species/petrel/petrel\_weightmatrix.txt ...

creating /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/config/species/petrel/petrel\_metapars.cfg ...

The necessary files for training petrel have been created.

Now, either run etraining or optimize\_parameters.pl with --species=petrel.

etraining quickly estimates the parameters from a file with training genes.

optimize\_augustus.pl alternates running etraining and augustus to find optimal metaparameters.

#make genbank

#gff2gbSmallDNA.pl LESP.gff3 ../droNov/training\_droNov.dna 1000 droNov.gb

gff2gbSmallDNA.pl LESP.gff3 /n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/genome.dna 1000 LESP.gb

#Randomly split the set of annotated sequences in a training and a test set

randomSplit.pl LESP.gb 200

#This generates a file myspecies.gb.test with 100 randomly chosen loci and a disjoint file myspecies.gb.train with the rest of the loci from genes.gb:

grep -c LOCUS LESP.gb\*

LESP.gb:9951

LESP.gb.test:200

LESP.gb.train:9751

#initial etraining

etraining --species=petrel LESP.gb.train

ls -ort $AUGUSTUS\_CONFIG\_PATH/species/petrel/

total 660

-rw-r--r-- 1 ysin 810 Mar 14 02:29 petrel\_weightmatrix.txt

-rw-r--r-- 1 ysin 7196 Mar 14 02:29 petrel\_parameters.cfg

-rw-r--r-- 1 ysin 1356 Mar 14 02:29 petrel\_metapars.utr.cfg

-rw-r--r-- 1 ysin 1445 Mar 14 02:29 petrel\_metapars.cgp.cfg

-rw-r--r-- 1 ysin 2057 Mar 14 02:29 petrel\_metapars.cfg

-rw-r--r-- 1 ysin 354227 Mar 14 02:48 petrel\_intron\_probs.pbl

-rw-r--r-- 1 ysin 261482 Mar 14 02:48 petrel\_exon\_probs.pbl

-rw-r--r-- 1 ysin 32489 Mar 14 02:48 petrel\_igenic\_probs.pbl

#Now we make a first try and predict the genes in genes.gb.train ab initio.

augustus --species=petrel LESP.gb.test | tee firsttest.out # takes ~1m

#Look at the accuracy report at the end of firsttest.out:

grep -A 22 Evaluation firsttest.out

\*\*\*\*\*\*\* Evaluation of gene prediction \*\*\*\*\*\*\*

---------------------------------------------\

| sensitivity | specificity |

---------------------------------------------|

nucleotide level | 0.879 | 0.786 |

---------------------------------------------/

----------------------------------------------------------------------------------------------------------\

| #pred | #anno | | FP = false pos. | FN = false neg. | | |

| total/ | total/ | TP |--------------------|--------------------| sensitivity | specificity |

| unique | unique | | part | ovlp | wrng | part | ovlp | wrng | | |

----------------------------------------------------------------------------------------------------------|

| | | | 686 | 514 | | |

exon level | 2425 | 2253 | 1739 | ------------------ | ------------------ | 0.772 | 0.717 |

| 2425 | 2253 | | 203 | 6 | 477 | 198 | 7 | 309 | | |

----------------------------------------------------------------------------------------------------------/

----------------------------------------------------------------------------\

transcript | #pred | #anno | TP | FP | FN | sensitivity | specificity |

----------------------------------------------------------------------------|

gene level | 289 | 200 | 30 | 259 | 170 | 0.15 | 0.104 |

----------------------------------------------------------------------------/

#optimize

optimize\_augustus.pl --species=petrel LESP.gb.train --cpus=12 1>petrel.stdout 2>petrel.stderr &

### optimize\_augustus.sh

##########################################################################

#!/bin/bash

#SBATCH -p general

#SBATCH --mem-per-cpu 8000

#SBATCH -n 12

#SBATCH -t 3-00:00:00

#SBATCH -o optimize\_augustus\_%A.out

#SBATCH -e optimize\_augustus\_%A.err

#SBATCH -J optimize\_augustus

#SBATCH --mail-type=ALL # Type of email notification- BEGIN,END,FAIL,ALL

#SBATCH --mail-user=yungwasin@fas.harvard.edu # Email to which notifications will be sent

source new-modules.sh

module load gcc/5.2.0-fasrc01 openmpi/2.0.1-fasrc01 maker/2.31.8-fasrc01

export AUGUSTUS\_CONFIG\_PATH=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/train\_augustus/config

optimize\_augustus.pl --species=petrel LESP.gb.train --cpus=12

##########################################################################

sbatch optimize\_augustus.sh

##retrain Augustus

#etraining --species=petrel LESP.gb.train

#check the prediction accuracy

augustus --species=petrel LESP.gb.test | tee secondtest.out

nano maker\_opts.ctl

And set:

snaphmm=/n/regal/edwards\_lab/petrel/RNA\_seq/all\_data/all\_rcorrector\_data/all\_data\_trimmed\_Wafergen/maker/LESP\_genome.maker.output/trainsnap/LESP1.hmm

est2genome=0

protein2genome=0

cp LESP\_genome.fasta LESP\_genome2.fasta

# and set genome=LESP\_genome2.fasta

# Or set the -base when run MAKER

maker -base LESP\_genome2