Week3-4: Annotation of genes with the MAKER Pipeline

Overview

MAKER is a powerful genome annotation pipeline designed for eukaryotic genomes. It integrates multiple sources of evidence—ab initio prediction models, RNA-Seq data, and protein homology—to generate comprehensive gene annotations. This tutorial outlines how to set up and run MAKER for genome annotation, followed by filtering, functional annotation and quality control.

1. **Ab Initio Gene Prediction**:
    - Tools like **Augustus** and **GeneMark** predict gene structures based on sequence patterns (e.g., start codons, splice sites).
2. **RNA-Seq Data**:
    - Provides direct evidence of gene expression, improving the accuracy of gene models predicted by ab initio methods.
3. **Protein Homology-Based Annotation**:
    - Uses known protein sequences from related species to predict gene structures.
    - Tools like **BLAST**, **Exonerate**, and **GenomeThreader** identify conserved genes by aligning transcripts and known proteins to the genome.
4. **Final Gene Model Refinement**:
    - Manual curation or additional refinement ensures biologically accurate annotations, producing high-quality gene models.

Steps to Perform Gene Annotation with MAKER

1. Create a Dedicated Directory

Start by creating a new directory where all annotation-related files will be stored. This will help keep your workspace organized.

```
WORKDIR=/path/to/gene_annotation_directory
mkdir -p $WORKDIR
cd $WORKDIR
```

2. Create Control Files

After navigating to your annotation directory, you need to generate the control files required for MAKER.

```bash
apptainer exec --bind $WORKDIR \
/data/courses/assembly-annotation-
course/CDS_annotation/containers/MAKER_3.01.03.sif maker -CTL
```

3. Edit the Control File

Open the `maker_opts.ctl` file in a text editor and modify the following parameters:

```
    #-----Genome (these are always required)
    genome=/path/to/<your genome assembly> #genome sequence (fasta file or
fasta embeded in GFF3 file)

    #-----EST Evidence (for best results provide a file for at least one)
    est=/path/to/<trinity transcriptome> #set of ESTs or assembled mRNA-
seq in fasta format. Use this for evidence based gene prediction

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

protein=/path/to/TAIR10_pep_20110103_representative_gene_model,/path/to/un
iprot-plant_reviewed.fasta #protein sequence file in fasta format (i.e.
from mutiple organisms). Use this for evidence based gene prediction

    #-----Repeat Masking (leave values blank to skip repeat masking)
    model_org= #select a model organism for DFam masking in RepeatMasker
**IMPORTANT!** switch if off
    rmlib=/path/to/$genome.mod.EDTA.TElib.fa #provide an organism specific
repeat library in fasta format for RepeatMasker
    repeat_protein=/path/to/PTREP20 #provide a fasta file of transposable
element proteins for RepeatRunner

    #-----Gene Prediction
    augustus_species=arabidopsis #Augustus gene prediction species model,
for ab-initio gene prediction
    est2genome=1 #infer gene predictions directly from ESTs, 1 = yes, 0 =
no
    protein2genome=1 #infer predictions from protein homology, 1 = yes, 0
= no

    #-----External Application Behavior Options
    cpus=1 #max number of cpus to use in BLAST and RepeatMasker. We will
run MAKER with MPI, so here we set it to 1

    #-----MAKER Behavior Options
    alt_splice=1 #Take extra steps to try and find alternative splicing, 1
= yes, 0 = no
    TMP=$SCRATCH
```

Pro tip: All necessary files are saved in `/data/courses/assembly-annotationcourse/CDS_annotation`. Either soft-link them to your annotation directory or specify the path to each file multiple transcriptome assemblies and protein databases can be provided. In this case, you will use your previously assembled RNA-Seq data (in fasta format), and proteins from A. thaliana and uniprot database.

For our run we will turn off the Repeatmasker libraries with `model_org=`, and use the EDTA TE library for Repeatmasker. It will save us a lot of time

> **NOTE!**
>
> Here, in th **augustus_species** parameter, we are using the pre-trained `arabidopsis` model from the Augustus database.
> If you have a new species or a species that is not present in the Augustus database, you have to train Augustus with your species-specific data.
> In that case, you can try **BRAKER** either with RNA-Seq or with protein data for training AUGUSTUS and GeneMark in a fully automated way.

4. Run MAKER with MPI

Example:

```
#!/bin/bash
#SBATCH --time=4-0
#SBATCH --mem=64G
#SBATCH -p pibu_el8
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=50
#SBATCH --job-name=Maker
#Sbatch --output=logs/Maker_gene_annotation_%j.out
#Sbatch --error=logs/Maker_gene_annotation_%j.err

COURSEDIR="/data/courses/assembly-annotation-course/CDS_annotation"
WORKDIR="/data/users/assembly-annotation-course"

REPEATMASKER_DIR="/data/courses/assembly-annotation-
course/CDS_annotation/softwares/RepeatMasker"
export PATH=$PATH:"/data/courses/assembly-annotation-
course/CDS_annotation/softwares/RepeatMasker"

module load OpenMPI/4.1.1-GCC-10.3.0
module load AUGUSTUS/3.4.0-foss-2021a

mpiexec --oversubscribe -n 50 apptainer exec \
--bind $SCRATCH:/TMP --bind $COURSEDIR --bind $AUGUSTUS_CONFIG_PATH --bind
$REPEATMASKER_DIR \
${COURSEDIR}/containers/MAKER_3.01.03.sif \
maker -mpi --ignore_nfs_tmp -TMP /TMP maker_opts.ctl maker_bopts.ctl
maker_evm.ctl maker_exe.ctl
```

5. Output Preparation

After running MAKER, check the output directory (usually named `datastore`). This directory contains subfolders for each individual contig from your genomic FASTA file. The `datastore_index.log` file will provide insights into the run status and where the data for individual contigs is stored.

To merge the individual GFF files into a single file, use the following commands:

```
   MAKERBIN="$COURSEDIR/softwares/Maker_v3.01.03/src/bin"
   $MAKERBIN/gff3_merge -s -d
assembly.maker.output/assembly_master_datastore_index.log >
assembly.all.maker.gff
   $MAKERBIN/gff3_merge -n -s -d
assembly.maker.output/assembly_master_datastore_index.log >
assembly.all.maker.noseq.gff
   $MAKERBIN/fasta_merge -d
assembly.maker.output/assembly_master_datastore_index.log -o assembly
```

> **Guiding questions:**
>
> - How many gene models were predicted by MAKER in your genome?
> - Is it comparable between accessions in the group and in the reference Arabidopsis thaiana genome?

---

6. Filtering and Refining Gene Annotations

1. Rename Genes and Transcripts

Create a directory to store the final filtered annotations and copy the necessary files to it:

```
mkdir final


protein="assembly.all.maker.proteins.fasta"
transcript="assembly.all.maker.transcripts.fasta"
gff="assembly.all.maker.noseq.gff"

cp $gff final/${gff}.renamed.gff
cp $protein final/${protein}.renamed.fasta
cp $transcript final/${transcript}.renamed.fasta

cd final
```

To assign clean, consistent IDs to the gene models, use MAKER's ID mapping tools.

```
$MAKERBIN/maker_map_ids --prefix $prefix --justify 7 ${gff}.renamed.gff >
id.map
$MAKERBIN/map_gff_ids id.map ${gff}.renamed.gff
$MAKERBIN/map_fasta_ids id.map ${protein}.renamed.fasta
$MAKERBIN/map_fasta_ids id.map ${transcript}.renamed.fasta
```

These commands update both the GFF3 and FASTA files, ensuring that all gene models have consistent, clean IDs. Here PREFIX is the a 3-4 letter prefix for your accession.

2. Run InterProScan on the Protein File

First, you will run **InterProScan** to annotate your protein sequences with functional domains, such as those from the Pfam database.

```
apptainer exec \
    --bind $COURSEDIR/data/interproscan-5.70-
102.0/data:/opt/interproscan/data \
    --bind $WORKDIR \
    --bind $COURSEDIR \
    --bind $SCRATCH:/temp \
    $COURSEDIR/containers/interproscan_latest.sif \
    /opt/interproscan/interproscan.sh \
    -appl pfam --disable-precalc -f TSV \
```

```
    --goterms --iprlookup --seqtype p \
    -i ${protein}.renamed.fasta -o output.iprscan
```

- Here we uses **Pfam** as the primary application in InterProScan. Additional applications like CDD, PANTHER, TIGRFAM, or SUPERFAMILY can be applied to the `-appl` flag. A full list of applications is available here.
- We are adding goterms and iprlookup to the output. These options provide additional functional annotations for the protein sequences.

3. Update GFF with InterProScan Results

Incorporate the InterProScan functional annotations into the GFF3 file using the `ipr_update_gff` tool.

```
$MAKERBIN/ipr_update_gff ${gff}.renamed.gff output.iprscan >
${gff}.renamed.iprscan.gff
```

4. Calculate AED Values

AED (Annotation Edit Distance) values are essential for evaluating how well gene models are supported by the evidence. Use MAKER's `AED_cdf_generator.pl` to generate AED values for all annotations.

```
perl $MAKERBIN/AED_cdf_generator.pl -b 0.025 ${gff}.renamed.gff >
assembly.all.maker.renamed.gff.AED.txt
```

After running this command, check whether most genes fall within the AED range of 0-0.5. This range indicates high confidence in the gene models.

- You can visualize the AED distribution using R.

5. Filter the GFF File for Quality

Filter the GFF file based on the AED values and/or functional annotations from InterProScan.

```
perl $MAKERBIN/quality_filter.pl -s ${gff}.renamed.iprscan.gff >
${gff}_iprscan_quality_filtered.gff
# In the above command: -s  Prints transcripts with an AED <1 and/or Pfam
domain if in gff3
```

6. Filter the GFF File for Gene Features

```
# We only want to keep gene features in the third column of the gff file
grep -P
"\tgene\t|\tCDS\t|\texon\t|\tfive_prime_UTR\t|\tthree_prime_UTR\t|\tmRNA\t
```

```
" ${gff}_iprscan_quality_filtered.gff > filtered.genes.renamed.gff3
# Check
cut -f3 filtered.genes.renamed.gff3 | sort | uniq
```

7. Extract mRNA Sequences and Filter FASTA Files

Extract the list of remaining mRNA IDs from the filtered GFF3 file, and use this list to filter the transcript and protein FASTA files:

```
module load UCSC-Utils/448-foss-2021a
grep -P "\tmRNA\t" filtered.genes.renamed.gff | awk '{print $9}' | cut -d
';' -f1 | sed 's/ID=//g' > list.txt
faSomeRecords ${transcript}.renamed.fasta list.txt
${transcript}.renamed.filtered.fasta
faSomeRecords ${protein}.renamed.fasta list.txt
${protein}.renamed.filtered.fasta
```

This step creates new FASTA files (`transcript.renamed.filtered.fasta` and `protein.renamed.filtered.fasta`) that only contain the sequences of high-quality gene models.

> **Guiding questions:**
>
> - How can you refine and validate gene annotations generated by MAKER?
> - What is the significance of the Annotation Edit Distance (AED) in assessing the quality of gene annotations?
> - Can functional annotations generated using tools like InterProScan help support the gene prediction?

Quality Assessment of Gene Annotations

1. **BUSCO: Quality Assessment of Gene Annotations**

BUSCO (Benchmarking Universal Single-Copy Orthologs) assesses genome or annotation completeness by identifying the presence of highly conserved single-copy orthologs across a broad range of taxa.

Step-by-Step Instructions

Run BUSCO on MAKER Annotations

You can run BUSCO on your MAKER-produced longest protein sequences (`${protein}.renamed.longest.fasta`) or longest transcript sequences (`${transcript}.renamed.longest.fasta`).

> **DIY: Extract Longest Protein and Transcript per Gene before running BUSCO**
>
> How to extract the longest protein and transcript per gene?
> You can get the length of each entry in a fasta file.
> Gene name is everything before `-R`, whereas `-RA`, `-RB`, `-RC` etc. are different isoforms

```
module load BUSCO/5.4.2-foss-2021a
busco -i maker_proteins.fasta -l brassicales_odb10 -o busco_output -m
proteins
busco -i maker_transcripts.fasta -l brassicales_odb10 -o busco_output -m
transcriptome
```

Here:

- `-i` specifies the input file (MAKER proteins or transcripts).
- `-l` specifies the lineage dataset (e.g., embryophyta).
- `-o` sets the output directory name.
- `-m` sets the mode to run on protein data (`proteins`, or `transcriptome` if using transcript sequences).

2. **Interpret BUSCO Results**

   The output provides several key metrics:

   - **Complete**: Number of BUSCO orthologs that are fully present in the annotation.
   - **Duplicated**: Number of duplicated BUSCO orthologs (it could indicate potential problems in assembly or polyploidy, however if you **forgot to take the longest protein or transcripts**, what else it could suggest?).
   - **Fragmented**: Partial BUSCO orthologs.
   - **Missing**: Orthologs that are not found.

   A high proportion of **complete** BUSCOs indicates a high-quality annotation.

3. **AGAT: Annotation Statistics**

   Use AGAT to generate comprehensive statistics about your gene annotations.

```
agat_sp_statistics.pl -i filtered.genes.renamed.gff -o annotation.stat
```

> **Guiding questions:**
>
> - What are the key metrics provided by BUSCO for assessing gene annotation quality for your dataset?
> - Can you visualize the Gene annotation tract with TE annotations from EDTA using circos or R packages?
> - How does the Gene density compare to the TE density along the scaffolds?

Visualizing Gene annotation with Geneious

Geneious Prime is a commercial, user-friendly desktop application for sequence analysis and visualization (Mac, Windows, Linux). Useful for interactive inspection of assemblies, annotations, alignments and read mappings.
You do not need the paid version to just visualize annotations.

There are other softwares as well that can be used to visualize: Jbrowse2 https://jbrowse.org/jb2/download/, or IGV https://igv.org/doc/desktop/

Get Geneious

Download and install from https://www.geneious.com. Start Geneious on your laptop.

Load a genome and set reference

- Drag and drop your genome FASTA file into Geneious to import.

Add annotation and alignment tracks

- Drag and drop GFF3 files onto the reference sequence to add gene annotations.

You can add multiple annotation tracks (e.g., **genes**, **TEs**) for comparison.

- Drag and drop Maker annotation with evidence transcript/protein tracks to visualize evidence used for gene prediction. This file is named as `${gff}_iprscan_quality_filtered.gff`

You can also map RNAseq bam files to the genome and visualize the expression of the genes, but not covered in this tutorial:

- BAM: File > Import > From File... (choose BAM).

Viewing and customizing

- Use the Zoom controls to inspect exon/intron structure and nucleotide sequence.
- Right-click tracks to change display options (color, show labels, collapse/expand).

- In the side pannel you can toggle which track to view and which to hide