Genome annotation procedure

Table of contents

1 Overview

After repeat masking (Section 1.1), each genome was annotated using three types of evidence: Illumina RNA-Seq (Section 1.2), PacBio Iso-Seq (Section 1.3), and OrthoDB Arthropoda proteins (Section 1.4). Gene models from all methods were then combined (Section 1.5), and functionally annotated (Section 1.6). Finally, tRNAs were predicted (Section 1.7).

1.1 Repeat library construction and repeat masking

```
RepeatModeler -database ${DB} -engine ncbi -pa 16
RepeatMasker -dir . -gff -u -no_is -xsmall -e ncbi -lib ${RMLIB} \
  -pa 16 genome.fasta
```

1.2 Gene prediction using Illumina RNA-Seq data

```
braker.pl --genome=${genome} \
  --rnaseq_sets_ids="rnaseq" \
  --rnaseq_sets_dirs=${rna} \
  --workingdir=${wdir} \
  --gff3 \
  --threads=16 \
  --verbosity=3 \
  --nocleanup --species=species_name
```

1.3 Gene prediction using PacBio Iso-Seq data

```
# Map IsoSeq to genome
minimap2 -t 16 -ax splice:hq ${GENOME} ${ISOSEQ} > long_reads.sam
samtools sort -O BAM -@ 16 long_reads.sam -o long_reads_sorted.bam

# Collapse isoforms
~/cDNA_Cupcake/cupcake/tofu/collapse_isoforms_by_sam.py \
  --input ${ISOSEQ} -b long_reads_sorted.bam --dun-merge-5-shorter -o isoseq

# Predict genes with GeneMarkS-T
~/Augustus/scripts/stringtie2fa.py -g ${GENOME} -f isoseq.collapsed.gff \
  -o isoseq.collapsed
${GENEMARK}/gmst.pl --strand direct isoseq.collapsed.mrna --output gmst.out \
  --format GFF
~/BRAKER/scripts/gmst2globalCoords.py -t isoseq.collapsed.gff -p gmst.out \
  -o gmst.global.gtf -g ${GENOME}
```

1.4 Gene prediction using OrthoDB protein data

```
galba.pl --genome=${genome} \
  --prot_seq=${protein} \
  --workingdir=${wdir} \
  --gff3 \
  --threads=16 \
  --verbosity=3 \
  --nocleanup --species=species_name
```

1.5 Integrating and formatting gene predictions

```
# create config file
echo -e "P 3\nE 0.15\nC 15\nM 0.5\nL 0.5\nintron_support 1\nstop_support 1\n\
start_support 2\ne_1 0\ne_2 1\ne_3 1\ne_4 300\e_5 50\ne_6 20" > long_reads.cfg

# run tsebra
python ~/TSEBRA/bin/tsebra.py -g ${BRAKER_GENES},${GALBA_GENES} \
  -e ${BRAKER_HINTS},${GALBA_HINTS} -l ${ISOSEQ_GENES} -c long_reads.cfg \
  -f -o tsebra_rm_short.gtf

# rename genes
python ~/TSEBRA/bin/rename_gtf.py --gtf tsebra_rm_short.gtf \
  --out tsebra_rm_short_renamed.gtf --translation_tab name_lookup.txt

# convert to gff3
~/Augustus/scripts/gtf2gff.pl < tsebra_rm_short_renamed.gtf \
  --out tsebra_rm_short_renamed.gff --gff3

# fix overlaps
~/agat/bin/agat_sp_fix_overlaping_genes.pl -f tsebra_rm_short_renamed.gff \
  -o tsebra_rm_short_renamed_nooverlap.gff
```

1.6 Functional annotation

```
nextflow run ~/NBIS/pipelines-nextflow \
  -profile singularity,nbis \
  -params-file params.yml \
  -c custom.config \
  -with-report report.html -with-trace \
  -resume
```

params.yml

```
subworkflow: 'functional_annotation'
genome: 'genome.fa'
gff_annotation: 'tsebra_rm_short_renamed_nooverlap.gff'
blast_db_fasta: 'uniprot_sprot.fasta'
```

```
outdir: 'results'
merge_annotation_identifier : 'ANNOTATION_ID'
```

1.6.1 Remove single-exon genes without functional annotation

```
# select all single-exon genes
~/agat/bin/agat_sp_filter_gene_by_intron_numbers.pl --gff ${GFF} \
  -o singleexon.gff

# remove single-exon genes without an InterPro domain
cut -f1 ${FUNCTIONAL_ANNOTATION}/InterPro.txt > interpro_names.txt
~/agat/bin/agat_sp_filter_feature_from_keep_list.pl --gff singleexon.gff \
  --kl interpro_names.txt -o singleexon_w_interpro.gff

# return single-exon genes with functional annotation to the remaining genes
agat_sp_merge_annotations.pl --gff singleexon_remaining.gff \
  --gff singleexon_w_interpro.gff -o ${NAME}_functional.gff
```

1.7 tRNA prediction

```
tRNAscan-SE -E --gff ${LABEL}_trnas.gff --thread 16 ${GENOME}
```