Log: Genome Annotation Chrysomya megacephala

tags: `Genome Annotation` `Cmeg`

Table of Contents

Genome annotation workflow:

- [x] 1- BUSCO
- [x] 2- QUAST
- [x] 3- Mithocondrial genome
- [x] 4- RepeatModeler
- [ ] 5- RepeatMasker
- [x] 6- RNA-seq
- [x] 7- Trinity
- [x] 8- STAR
- [ ] 9- Braker

https://github.com/CBC-UCONN/Genome_Assembly
https://github.com/CBC-UCONN/Structural-Annotation

List of software:

- [x] QUAST (/usr/local/bin/quast.py) (Pedro!)
- [x] RepeatModeler (/RepeatModeler-2.0.4)
- [x] RepeatMasker(/dados/home/pedro/Programs/RepeatMasker/RepeatMasker)
- [x] STAR (/dados/home/bruno/anaconda3/bin/STAR)
- [x] samtools (/dados/home/bruno/anaconda3/bin/samtools
  /dados/home/bruno/anaconda3/bin/samtools.pl)
- [x] Braker (docker image ID: 7772eca57cee)
    - https://hub.docker.com/r/teambraker/braker3
- [x] TSEBRA (/)
    - https://github.com/Gaius-Augustus/TSEBRA
- [x] AUGUSTUS (docker image ID: c0dfd27799fc)
    - https://github.com/Gaius-Augustus/Augustus
- [x] busco (/usr/local/bin/busco)
- [x] gFACS (/)
    - https://gitlab.com/PlantGenomicsLab/gFACs
- [x] EnTAP (/)

- https://entap.readthedocs.io/en/v0.8.0-beta/introduction.html

- [x] GeneMark-ETP (PEDRO!)
  - https://github.com/gatech-genemark/GeneMark-ETP

Genome sequencing

We sequenced 1 male at Dovetail (Pac-Bio).

ANOTAR AQUI DADOS DO SEQUENCIAMENTO

Number of reads 2521875
Coverage (x) 66
HPA Length (bp) 1446676375
HPA N50 (bp) 1148833
HPA N90 (bp) 117041
HPA L50 287
HPA L90 1800
FA Length (bp) 671207201
FA N50 (bp) 2214294
FA N90 (bp) 540686
FA L50 84
FA L90 316

bp = Base pair; HPA = Hifiasm (Cheng et al., 2021) primary assembly; FA = Final assembly. N50 = Sequence length of the smallest contig within those that sum up to 50% of the total genome's length; N90 =Sequence length of the smallest contig within those that sum up to 90% of the total genome's length; L50 = Smallest sequence number that together sum 50% of the total genome's length; L90 = Smallest sequence number that together sum 90% of the total genome's length.

1- BUSCO First

Busco version 5.3.2

```
cd /home/blowflies/genome_annotation/cmeg/1-busco_first

sudo docker pull ezlabgva/busco:v5.4.4_cv1

cp ../0-genome/cmeg.fa .

sudo docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.4.4_cv1 bus
```

2- QUAST

Quast accepts assemblies and reference genomes in FASTA format. Files may be compresses with zip, gzip or bzip2.
Quast accepts Illumina, PacBio, and Oxford Nanopore reads in FASTQ format (may be compressed).

Running

```
pwd
/home/blowflies/genome_annotation/cmeg/2-quast

# OLD
# /quast-5.0.2/quast.py ../0-genome/purged.fa -t 20 --eukaryote --large --rna

# Now we're running from the python library

# Obs: had to edit the jsontemplate.py script again to change cgi to html
# We're in the root
sudo find / -type d -name "quast_libs"

# the script we need to edit is here:
/usr/local/lib/python3.8/dist-packages/quast-5.0.2-py3.8.egg/quast_libs/site_

# Running
sudo quast.py ../0-genome/purged.fa -t 20 --eukaryote --large --rna-finding
```

View

`less quast_results/latest/report.txt`

3- Mitochondrial genome

Cmeg mitochondrial genome on NCBI:
https://www.ncbi.nlm.nih.gov/nucleotide/NC_019633.1
NC_019633.1

```
cd home/blowflies/genome_annotation/cmeg/3-mitochondrial_genome
mkdir 1-blastn
# We downloanded mitochondrial sequence from NCBI to a local computer and tra
```

Blastn

version 2.11

```
cd home/blowflies/genome_annotation/cmeg/3-mitochondrial_genome/1-blastn
#Making database using the genome
makeblastdb -in ../Cmeg_ref_mitochondria.fasta -dbtype nucl -out cmeg_mit_dat
#Running
blastn -task blastn -evalue 0.00001 -db ./cmeg_mit_database -query ../../0-ge
```

The best aligment was against scaffold 268, which is a very big scaffold (1.9Mb). We isolated this scaffold and cut it in three pieces "before mitochondria" - "mitochondria" - "after mitochondria"

```
cd /home/blowflies/genome_annotation/cmeg/0-genome

# Remove scaffold from genome
seqkit grep -v -p "ptg000268l" cmeg.fa > cmeg_N_genome.fa

# Isolate scaffold
seqkit grep -p "ptg000268l" cmeg.fa > ptg000268l.fa

#Isolating the motochondria
seqkit subseq ptg000168l.fa -r 1324186:1339448 > cmeg_mit.fa

#Removing the mitochondria and breaking the scaffold in two
seqkit  subseq ptg000268l.fa -r 1:1305525 > cmeg_ptg268l_1.fa
seqkit  subseq ptg000268l.fa -r 1341184:1921755 > cmeg_ptg268l_2.fa

#Renaming scaffold ids
sed -i 's/>ptg000268l/>ptg000268l_1/' cmeg_ptg268l_1.fa
sed -i 's/>ptg000268l/>ptg000268l_2/' cmeg_ptg268l_2.fa
sed -i 's/>ptg000268l/>cmeg_mit/' cmeg_mit.fa

# Joining scaffolds to genome again
cat cmeg_N_genome.fa cmeg_ptg268l_1.fa cmeg_ptg268l_2.fa > cmeg_N_genome_fina

#Checking
grep ">ptg000268l" cmeg_N_genome_final.fa
grep -c ">" cmeg_N_genome_final.fa
```

MITOS Web Server

We ran the mithocondrial genome annotation using MITOS2 web server with all the default parameters but the genetic code, which was specified to be the invertebrate one.

We then downloaded the output files to a local computer and sent them to 2-MITOS_results (/home/blowflies/genome_annotation/cmeg/3-mitochondrial_genome/2-MITOS_results).

The job settings were:

Job ID: cmeg

Property	Value
Reference	RefSeq 63 Metazoa
Genetic Code	5
Proteins	True
tRNAs	True
rRNAs	True
OH	True
OL	True
Circular	True
Use Al Arab et al.	False
E-value Exponent	2.0
Final Maximum Overlap	50nt
Fragment Quality Factor	100.0
Standard Code	False
Cutoff	50.0%
Clipping Factor	10.0
Fragment Overlap	20.0%
Local only	True
Sensitive only	False
ncRNA overlap:	50 nt

TEST on pacbio genome before assembly

```
/home/blowflies/genome_annotation/cmeg/3-mitochondrial_genome
mkdir 2-blast_pacbio
# we copied the fastq with the raw pacbio sequences to 2-blast_pacbio
gzip -d cmeg_raw_genome.fastq.gz
#converting fastq to fasta
seqkit fq2fa cmeg_raw_genome.fastq -o cmeg_raw_genome.fa
#blastn
blastn -task blastn -evalue 0.00001 -db ../1-blastn/cmeg_mit_database -query
```

Pairwise Sequence Alignment Tools (Reverse complement + Trimming the mt genome)

```
#We are here
cd /home/blowflies/genome_annotation/cmeg/0-genome

#First rough trimming
seqkit  subseq ptg000268l.fa -r 1305525:1341184 > cmeg_mit_repeats.fa

#Downloaded to local computer
scp -P 2205 diniz@143.107.244.181:/home/blowflies/genome_annotation/cmeg/0-g
```

https://www.ebi.ac.uk/Tools/psa/

- Water (EMBOSS)
  EMBOSS Water uses the Smith-Waterman algorithm (modified for speed enhancements) to calculate the local alignment of two sequences.

- Stretcher (EMBOSS)
  EMBOSS Stretcher uses a modification of the Needleman-Wunsch algorithm that allows larger sequences to be globally aligned.

- Matcher (EMBOSS)
  EMBOSS Matcher identifies local similarities between two sequences using a rigorous algorithm based on the LALIGN application.

4- RepeatModeler

Couldn't run without docker!!! I don't know what is happening (Vanessa)

Installing with docker

It was complicated installing all the programs, so we used a Docker container

```
#Always use this before repeat modeler
docker run -it --rm dfam/tetools:latest

container-ID: 1b8109bcf5aa
container-name: hungry_banach
container-image: dfam/tetools:latest

#To attach the container and continue running press CTRL+P, then CTRL+Q
```

Creating database

```
# Moving the fasta file to docker container from the Rosalind server using:
# docker cp file.txt container-name:/path/to/copy/file.txt
docker cp ./cmeg_N_genome_final.fa hungry_banach:/home/cmeg

# Getting inside the container
docker exec -it hungry_banach /bin/bash

#Database
BuildDatabase -name cmeg_database cmeg_N_genome_final.fa
```

Running

```
RepeatModeler -database cmeg_database -threads 30 -LTRStruct  >log 2>err

#Then to exit the container and continue running press CTRL+C
```

Copying the results

```
# Compressing files inside docker
tar -cjvf cmeg_other_files.tar.gz *

# now here
cd /home/cunha/01-RepeatModeler/cmeg

docker cp -a hungry_banach:/home/cmeg/cmeg_database-families.stk ./
docker cp -a hungry_banach:/home/cmeg/cmeg_database-families.fa ./
docker cp -a hungry_banach:/home/cmeg/cmeg_database-rmod.log ./
docker cp -a hungry_banach:/home/cmeg/log ./
docker cp -a hungry_banach:/home/cmeg/err ./
docker cp -a hungry_banach:/home/cmeg/cmeg_other_files.tar.gz ./
```

Test - LTR Harvest

First, we split the genome, to see if it works better

```
cd /home/blowflies/genome_annotation/cmeg/0-genome

seqkit split2 -p 5 cmeg_N_genome_final.fa

cd cmeg_N_genome_final.fa.split/

for i in *; do mv $i cmeg_${i##cmeg_N_genome_final.part_00}; done
```

file	format	type	num_seqs	sum_len	min_len	avg_len	max_len
cmeg_1.fa	FASTA	DNA	152	118,059,414	11,549	776,706.7	8,638,607
cmeg_2.fa	FASTA	DNA	152	159,414,450	13,769	1,048,779.3	9,257,387
cmeg_3.fa	FASTA	DNA	152	142,440,787	12,578	937,110.4	14,378,473
cmeg_4.fa	FASTA	DNA	152	137,312,730	11,578	903,373.2	7,470,457
cmeg_5.fa	FASTA	DNA	152	113,944,162	11,647	749,632.6	6,742,468

Now, running LTRHarvest

```
# Doing it on Rosalind
# Container: cranky_morse

cd /home/blowflies/genome_annotation/cmeg/0-genome/cmeg_N_genome_final.fa.sp

for i in *; do docker cp $i cranky_morse:/home; done

docker exec -it cranky_morse /bin/bash

cd home/
mkdir 1 2 3 4 5
mv *1.fa 1/
mv *2.fa 2/
mv *3.fa 3/
mv *4.fa 4/
mv *5.fa 5/

# LTR.sh
cd /home/1
/opt/genometools/bin/gt suffixerator -db cmeg_1.fa -indexname cmeg_1_db -tis
/opt/genometools/bin/gt ltrharvest -index cmeg_1_db -minlenltr 100 -maxlenlt
/opt/LTR_retriever/LTR_retriever -genome cmeg_1.fa -inharvest cmeg_1.harvest

cd /home/2
/opt/genometools/bin/gt suffixerator -db cmeg_2.fa -indexname cmeg_2_db -tis
/opt/genometools/bin/gt ltrharvest -index cmeg_2_db -minlenltr 100 -maxlenlt
/opt/LTR_retriever/LTR_retriever -genome cmeg_2.fa -inharvest cmeg_2.harvest

cd /home/3
/opt/genometools/bin/gt suffixerator -db cmeg_3.fa -indexname cmeg_3_db -tis
/opt/genometools/bin/gt ltrharvest -index cmeg_3_db -minlenltr 100 -maxlenlt
/opt/LTR_retriever/LTR_retriever -genome cmeg_3.fa -inharvest cmeg_3.harvest

cd /home/4
/opt/genometools/bin/gt suffixerator -db cmeg_4.fa -indexname cmeg_4_db -tis
/opt/genometools/bin/gt ltrharvest -index cmeg_4_db -minlenltr 100 -maxlenlt
/opt/LTR_retriever/LTR_retriever -genome cmeg_4.fa -inharvest cmeg_4.harvest

cd /home/5
/opt/genometools/bin/gt suffixerator -db cmeg_5.fa -indexname cmeg_5_db -tis
/opt/genometools/bin/gt ltrharvest -index cmeg_5_db -minlenltr 100 -maxlenlt
/opt/LTR_retriever/LTR_retriever -genome cmeg_5.fa -inharvest cmeg_5.harvest

cd /home
```

It worked! Now we need to download the final files back from the container, concatenate them with the output from the previous RepeatModeler run, and then use this final file to mask the genome

```
cd /home/blowflies/genome_annotation/cmeg/4-RepeatModeler
# cmeg_database-families.fa is here

docker cp -a cranky_morse:/home/1/cmeg_1.fa.LTRlib.fa ./
docker cp -a cranky_morse:/home/2/cmeg_2.fa.LTRlib.fa ./
docker cp -a cranky_morse:/home/3/cmeg_3.fa.LTRlib.fa ./
docker cp -a cranky_morse:/home/4/cmeg_4.fa.LTRlib.fa ./
docker cp -a cranky_morse:/home/5/cmeg_5.fa.LTRlib.fa ./

docker cp -a cranky_morse:/home/cmeg_LTR.tar.gz ./ # other outputs

cat *fa > cmeg_modeler_complete.fa

# remove redundancy
vsearch --cluster_fast cmeg_modeler_complete.fa -id 0.80 -threads 15 -centro
```

5- RepeatMasker

Installing steps can be found in the *Chrysomya megacephala* log (https://hackmd.io/ziOztK1MQZecVm0_qDAr1g)

##Doing it on Rosalind

```
# from Darwin to Rosalind
cd /home/cunha/01-RepeatModeler/cmeg

scp -P 2205 * pedro@143.107.244.181:/home/blowflies/genome_annotation/cmeg/4-

# running
cd /home/blowflies/genome_annotation/cmeg/5-RepeatMasker

RepeatMasker -lib /home/blowflies/genome_annotation/cmeg/4-RepeatModeler/cmeg
```

5.1 BUSCO

```
cd /home/blowflies/genome_annotation/cmeg/5-RepeatMasker
mkdir 1-BUSCO
cp cmeg_N_genome_final.fa.masked 1-BUSCO/
cd 1-BUSCO
sudo docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.4.7_cv1 bus
```

Results

```
---------------------------------------------------
|Results from dataset diptera_odb10               |
---------------------------------------------------
|C:97.8%[S:86.6%,D:11.2%],F:0.2%,M:2.0%,n:3285    |
|3214   Complete BUSCOs (C)                       |
|2846   Complete and single-copy BUSCOs (S)       |
|368    Complete and duplicated BUSCOs (D)        |
|8      Fragmented BUSCOs (F)                     |
|63     Missing BUSCOs (M)                        |
|3285   Total BUSCO groups searched               |
---------------------------------------------------
```

5.2 QUAST

```
#comparing with the file /home/blowflies/genome_annotation/cmeg/0-genome/cmeg
cd /home/blowflies/genome_annotation/cmeg/5-RepeatMasker
mkdir 2-QUAST
cp cmeg_N_genome_final.fa.masked 2-QUAST/
cd 2-QUAST
mkdir cmeg.fa
cd cmeg.fa
sudo quast.py cmeg_N_genome_final.fa.masked -t 20 --eukaryote --large --rna-
```

```
#Results:
cd /home/blowflies/genome_annotation/cmeg/5-RepeatMasker/2-QUAST/cmeg.fa/quas
cat report.txt
```

```
All statistics are based on contigs of size >= 3000 bp, unl

Assembly                    cmeg_N_genome_final.fa.masked
# contigs (>= 0 bp)         760
# contigs (>= 1000 bp)      760
# contigs (>= 5000 bp)      760
# contigs (>= 10000 bp)     760
# contigs (>= 25000 bp)     703
# contigs (>= 50000 bp)     601
Total length (>= 0 bp)      671171543
Total length (>= 1000 bp)   671171543
Total length (>= 5000 bp)   671171543
Total length (>= 10000 bp)  671171543
Total length (>= 25000 bp)  670017351
Total length (>= 50000 bp)  666394117
# contigs                   760
Largest contig              14378473
Total length                671171543
Estimated reference length  500000000
GC (%)                      29.14
N50                         2214294
NG50                        2991992
N75                         1082448
NG75                        2010070
L50                         84
LG50                        50
L75                         190
LG75                        102
# total reads               2110
# left                      0
# right                     0
Mapped (%)                  100.0
Properly paired (%)         0.0
Avg. coverage depth         1
Coverage >= 1x (%)          99.95
# N's per 100 kbp           0.05
# predicted rRNA genes      340 + 125 part
```

6- RNA-seq

We extracted RNA from:

- 50 eggs
- 10 L1
- 5 L2
- 2 L3
- 1 pupae
- 1 virgin female
- 1 gravid female
- 1 male

Then, we pooled all the samples (2ug of RNA from each sample) and sequenced it.

- RNAseq Illumina 20M reads paired end PE150 Q30>85%

```
#Coping files
cp -r /home/Raw_seqs/cmeg_pool_RNA /home/blowflies/genome_annotation/cmeg/
#Renaming
cd /home/blowflies/genome_annotation/cmeg/
mv cmeg_pool_RNA 8-cmeg_pool_RNA
#checking md5
cd 8-cmeg_pool_RNA/
cat MD5.txt
#9befbbedbbd4e5d9b79a97d29eefc0be  Cmeg_1.fq.gz
#1be90c92ec0a0e97240fe0ce21d4a487  Cmeg_2.fq.gz
md5sum Cmeg*
#9befbbedbbd4e5d9b79a97d29eefc0be  Cmeg_1.fq.gz
#1be90c92ec0a0e97240fe0ce21d4a487  Cmeg_2.fq.gz
mkdir 0-raw_reads
mv *.fq.gz 0-raw_reads
mv MD5.txt 0-raw_reads
```

Quality control of raw reads

We ran FastQC and, then MultiQC.

> Don't need to unzip raw read files because fastqc can cope with zipped files (.gz).

FastQC will will process one sample at a time and give you an output report for each sample separately. MultiQC will combine all the outputs from FastQC analysis and give you one QC report for all processed samples, making them more easily comparable.
–> nice webpage on fastqc and multiqc: https://stab.st-andrews.ac.uk/wiki/index.php/FASTQC_and_MultiQC
–> https://multiqc.info/

```
cd /home/blowflies/genome_annotation/cput/4-cmeg_pool_RNA
mkdir 1-QC
cd 0-raw_reads
fastqc *fq.gz #v0.11.9

multiqc . #Version 1.11
mv *.html ../1-QC
mv *.zip ../1-QC
mv multiqc_data ../1-QC
```

Results: https://drive.google.com/file/d/1ggrh–DgYnWfA8WH2X4tdXRJ4Xo9ZHac/view?usp=share_link

Coping multiqc report to a local computer

```
scp -P 2205 vanessa@143.107.244.181:/home/blowflies/genome_annotation/cmeg/4-cmeg_pool_RNA/1-QC/multiqc_report.html /mnt/c/Users/vansc/Downloads
```

Trimming

Processing raw reads to trimming (remove only bad quality bases).

I used Trimmometic to trimming version 0.39
–> nice webpage on how to use Trimmomatics: http://www.usadellab.org/cms/index.php?page=trimmomatic
https://datacarpentry.org/wrangling-genomics/03-trimming/

```
cd /home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/0-raw-reads
mkdir ../2-trimming
screen
TrimmomaticPE Cmeg_1.fq.gz Cmeg_2.fq.gz -threads 8 -baseout /home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/cmeg.trimmed.fq.gz ILLUMINACLIP:TruSeq3-PE-2.fa:2:30:10 SLIDINGWINDOW:4:15 MINLEN:36
```

Quality control of trimmed reads

```
# in this directory -> /home/blowflies/genome_annotation/cmeg/4-cmeg_pool_RNA/2-tr

# QC
fastqc *.gz

# here -> /home/blowflies/genome_annotation/cmeg/4-cmeg_pool_RNA/1-QC
mkdir trimmed_reads_qc

# in this directory -> /home/blowflies/genome_annotation/cmeg/4-cmeg_pool_RNA/2-tr
mv *.html ../1-QC/trimmed_reads_qc/
mv *.zip ../1-QC/trimmed_reads_qc/
multiqc .
```

Observation

We renamed the genome file from this step onwards.

```
cd /home/blowflies/genome_annotation/cmeg/0-genome

mv purged.fa cmeg.fa
```

Quality Control of raw reads

```
fastqc /home/cunha/03-RNA/01-Reads/cmeg/sra_ncbi/*fastq

# in /home/cunha/03-RNA/01-Reads/cmeg/sra_ncbi/02-qc
multiqc .
```

Trimimng

```
#!/bin/bash

#SBATCH -N 1-10
#SBATCH -n 20
#SBATCH -t 240:00:00
#SBATCH -p long
#SBATCH -o cmeg_trimmomatic.out

srun /home/cunha/anaconda3/bin/trimmomatic PE SRR1660427_1.fastq SRR1660427_
srun /home/cunha/anaconda3/bin/trimmomatic PE SRR1663113_1.fastq SRR1663113_
srun /home/cunha/anaconda3/bin/trimmomatic PE SRR1663114_1.fastq SRR1663114_
```

7 - Trinity

We had to send the files to /home/cunha/03-RNA/01-Reads/cmeg directory and then rename all of them to make it easier to run trinity

```
#sending the files
rsync -av /home/cunha/Genotype_Phenotype/0-sequences/1-transcriptome/Cmeg/1-

#renaming them
# rename files
for f in *fastq.gz; do mv -- "$f" "${f%.fastq.gz}.fq.gz"; done
```

Running Trinity

```
#!/bin/bash

#SBATCH --job-name trinity_cmeg ## nome que aparecerá na fila
#SBATCH --output trinity_cmeg.out ## nome do arquivo de saída; o %j é igual a
#SBATCH --ntasks=1 ## número de tarefas (análises) a serem executadas
#SBATCH --cpus-per-task=20 ## o número de threads alocados para cada tarefa
#SBATCH --mem-per-cpu=1000M # memória por núcleo da CPU
#SBATCH --partition=long ## as partições a serem executadas (separadas por v
#SBATCH --time=10-00:00:00 ## hora para análise (dia-hora:min:seg)
#SBATCH --error=err

srun docker run --rm -v`pwd`:`pwd` trinityrnaseq/trinityrnaseq Trinity --seq
```

The final assembly is here: /home/cunha/03-RNA/02-Trinity (Darwin). And it was renamed to cmeg_trinity.fasta

Assembly statistics

```
file                          format  type  num_seqs      sum_len  min_len
cmeg_trinity_all.Trinity.fasta  FASTA   DNA    154,054  137,251,875      166
```

BUSCO (transcriptome quality)

```
mkdir /home/cunha/03-RNA/03-Busco/BUSCO_RNA_all

docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.4.7_cv1 busco -
```

8- STAR - RNAseq alignment

Index

> All genome FASTA files **cannot** be zipped

```
# unziping files
cd
gzip -dk *P*

# We need to create a directory where the genome indexes will be stored befo
cd /home/blowflies/genome_annotation/cmeg
mkdir 9-STAR_new
chmod 777 9-STAR_new
cd 9-STAR_new
mkdir star_index
chmod 777 star_index

STAR --runThreadN 6 --runMode genomeGenerate --genomeDir /home/blowflies/gen

#OLD
#STAR --runThreadN 6 --runMode genomeGenerate --genomeDir /home/blowflies/ge
```

Before the alignment itself we had to concatenate all the fastq files available. The files are in Rosalind (/home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all)

```
cd /home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all
cat *_1P.fq.gz > cmeg_all_1P.fq.gz
cat *_2P.fq.gz > cmeg_all_2P.fq.gz
```

STAR aligment

```
cd /home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all

for i in *_1P.fq.gz; do STAR --runMode alignReads --readFilesCommand zcat --

#OLD
#for i in *_1P.fq.gz; do
#STAR --runMode alignReads --readFilesCommand zcat --outSAMtype BAM SortedBy
```

Results:

```
(base) vanessa@c548:/home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all$ for i in *_1P.fq.gz; do STAR --runMode alignReads --readFilesCommand zca
t --outSAMtype BAM SortedByCoordinate Unsorted --genomeDir /home/blowflies/genome_annotation/cmeg/9-STAR_new/star_index --readFilesIn $i ${i%_1P.fq.gz}_2P.fq.gz --
runThreadN 20 --outFileNamePrefix /home/blowflies/genome_annotation/cmeg/9-STAR_new/${i%_1P.fq.gz}; done
        STAR --runMode alignReads --readFilesCommand zcat --outSAMtype BAM SortedByCoordinate Unsorted --genomeDir /home/blowflies/genome_annotation/cmeg/9-STAR_ne
w/star_index --readFilesIn cmeg_all_1P.fq.gz cmeg_all_2P.fq.gz --runThreadN 20 --outFileNamePrefix /home/blowflies/genome_annotation/cmeg/9-STAR_new/cmeg_all
        STAR version: 2.7.10a   compiled: 2022-01-14T18:50:00-05:00 :/home/dobin/data/STAR/STARcode/STAR.master/source
Mar 15 11:45:33 ..... started STAR run
Mar 15 11:45:33 ..... loading genome
Mar 15 11:46:26 ..... started mapping
Mar 15 14:27:21 ..... finished mapping
Mar 15 14:27:22 ..... started sorting BAM

EXITING because of fatal ERROR: not enough memory for BAM sorting:
SOLUTION: re-run STAR with at least --limitBAMsortRAM 14559922796
Mar 15 14:27:22 ...... FATAL ERROR, exiting
(base) vanessa@c548:/home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all$
```

Trying again:

```
#On Rosalind
cd /home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all
for i in *_1P.fq.gz; do STAR --runMode alignReads --readFilesCommand zcat --
```

It worked and finished successfully!!!

9- BRAKER3 - Structural annotation

We ran BRAKER3 for structural annotation.

"BRAKER3 is the latest pipeline in the BRAKER suite. It enables the usage of RNA-seq and protein data in a fully automated pipeline to train and predict highly reliable genes with GeneMark-ETP and AUGUSTUS. The result of the pipeline is the combined gene set of both gene prediction tools, which only contains genes with very high support from extrinsic evidence." (https://github.com/Gaius-Augustus/BRAKER)

We needed to copied the masked genome and the sorted bam to /home/diniz/programs/braker in Rosalind to run BRAKER3.

```
cd/home/diniz/programs/braker
mkdir Cmeg_braker_new
cd Cmeg_braker_new
cp /home/blowflies/genome_annotation/cmeg/5-RepeatMasker/cmeg_N_genome_final
md5sum cmeg_N_genome_final.fa.masked
#md5 checked
cp /home/blowflies/genome_annotation/cmeg/9-STAR_new/cmeg_allAligned.out.bam
md5sum cmeg_allAligned.out.bam
#9ffcfde10312b6eca9680911ad6f51ec
md5sum /home/blowflies/genome_annotation/cmeg/9-STAR_new/cmeg_allAligned.out
#9ffcfde10312b6eca9680911ad6f51ec
```

#OLD

STAR couldn't allocate RAM memory for bam sorting, so we did it manually with samtools

```
cd /home/blowflies/genome_annotation/cmeg/9-STAR_new/
samtools sort -o cmeg_allAligned.sort.out.bam cmeg_allAligned.out.bam
```

After that we moved the genome file Cmeg_masked.fasta (from /home/pedro/Non_Coding_Element_Evolution/3-Masking/2-RepeatMasker/) and the sorted bam to /home/diniz/braker in rosalind and ran BRAKER3. We ran the command line in a bash script (cmeg_annot.sh).

BRAKER3 run

```
# need to run first
export BRAKER_SIF=/home/diniz/programs/braker/braker3.sif
nohup bash cmeg_annot.sh &

# the content of cmeg_annot.sh
#New
singularity exec /home/diniz/programs/braker/braker3.sif braker.pl --genome=
#--skipOptimize


#OLD
#braker.pl --genome=/home/diniz/braker/cmeg_final_results/Cmeg_masked.fasta
```

Busco evaluation with protein sequences

```
cd /home/diniz/programs/braker/Cmeg_braker_new/

docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.4.4_cv1 busco -
```

Results:

```
---------------------------------------------------
|Results from dataset diptera_odb10               |
---------------------------------------------------
|C:97.1%[S:73.4%,D:23.7%],F:0.7%,M:2.2%,n:3285    |
|3188   Complete BUSCOs (C)                       |
|2410   Complete and single-copy BUSCOs (S)       |
|778    Complete and duplicated BUSCOs (D)        |
|23     Fragmented BUSCOs (F)                     |
|74     Missing BUSCOs (M)                        |
|3285   Total BUSCO groups searched               |
---------------------------------------------------
```

BRAKER3 second run

```
# need to run first
export BRAKER_SIF=/home/diniz/programs/braker/braker3.sif

cd /home/diniz/programs/braker/Cmeg_braker_new
mkdir second_run
cd second_run
nano cmeg_annot_2.sh
#cmeg_annot_2.sh
singularity exec /home/diniz/programs/braker/braker3.sif braker.pl --genome=
```

```
#Run
cd /home/diniz/programs/braker/Cmeg_braker_new/second_run
nohup bash cmeg_annot_2. &
```

How many transcripts

```
cd /home/diniz/programs/braker/Cmeg_braker_new/second_run
grep -c ">" braker.codingseq
#28019
```

Total transcripts: 28019

Busco evaluation with protein sequences from second Braker run

```
cd /home/diniz/programs/braker/Cmeg_braker_new/second_run
mv braker.gtf cmeg_braker.gtf
mv braker.aa cmeg_braker.aa
mv braker.codingseq cmeg_braker.codingseq
md5sum cmeg*
#410ad1b4fc14eba1faf5e19f2a1951ca  cmeg_braker.aa
#fc8e26a8c1721f6ba954e0e8c5fdbe9f  cmeg_braker.codingseq
#5b340dc2a5033183337f48b66aac6926  cmeg_braker.gtf


docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.4.4_cv1 busco -
```

Results:

```
---------------------------------------------------
|Results from dataset diptera_odb10               |
---------------------------------------------------
|C:97.0%[S:78.6%,D:18.4%],F:0.7%,M:2.3%,n:3285    |
|3185   Complete BUSCOs (C)                       |
|2581   Complete and single-copy BUSCOs (S)       |
|604    Complete and duplicated BUSCOs (D)        |
|24     Fragmented BUSCOs (F)                     |
|76     Missing BUSCOs (M)                        |
|3285   Total BUSCO groups searched               |
---------------------------------------------------
```

Coping all braker files to /home/blowflies/genome_annotation/cmeg

```
cd /home/blowflies/genome_annotation/cmeg
mkdir 10-Braker_new
cd 10-Braker_new
sudo cp -r /home/diniz/programs/
braker/Cmeg_braker_new ./
#md5 checked
```

11- EnTAP

We did it on Darwin using the output from the restart Bracker run

```
#We copied the aminoacid file from Rosalind (/home/diniz/programs/braker/Cmeg
ssh martins@lem.ib.usp.br -p 4988
55rx64yz$
cd /home/martins/EnTAP_restart/cmeg/
mkdir new
cd new

#copying new file from Rolalind to Darwin server
scp -P 2205 diniz@143.107.244.181:/home/diniz/programs/braker/Cmeg_braker_new
#Checking md5
#md5 on Darwin
md5sum cmeg_braker.aa
#410ad1b4fc14eba1faf5e19f2a1951ca
#md5 on Rosalind
#410ad1b4fc14eba1faf5e19f2a1951ca

#Running EnTAP
#!/bin/bash

#SBATCH --job-name entap_cmeg ## nome que aparecerá na fila
#SBATCH --output entap_cmeg.out ## nome do arquivo de saída; o %j é igual a j
#SBATCH --ntasks=1 ## número de tarefas (análises) a serem executadas
#SBATCH --cpus-per-task=10 ## o número de threads alocados para cada tarefa
#SBATCH --mem-per-cpu=1000M # memória por núcleo da CPU
#SBATCH --partition=long ## as partições a serem executadas (separadas por v
#SBATCH --error=err

srun EnTAP --runP -i /home/martins/EnTAP_restart/cmeg/new/cmeg_braker.aa -d /


#checking md5
md5sum entap_outfiles/final_results/*
#d41d8cd98f00b204e9800998ecf8427e  annotated_contam.faa
#9fc1ae9cc0246a8202519b62f903e047  annotated_contam_gene_ontology_terms.tsv
#12da35de58c8400536adef688be44687  annotated_contam.tsv
#b89d36011f0929530fa3b7b76ce0d1d7  annotated.faa
#0e2f85b9015bb174fd47329979650832  annotated_gene_ontology_terms.tsv
#4438ac6c503707545294f5b43d14cde7  annotated.tsv
#b89d36011f0929530fa3b7b76ce0d1d7  annotated_without_contam.faa
#0e2f85b9015bb174fd47329979650832  annotated_without_contam_gene_ontology_te
#4438ac6c503707545294f5b43d14cde7  annotated_without_contam.tsv
#1a8f2d7eac09b46de2802807c0853d2b  entap_results.tsv
#1e1bd41573cd19d2704e67bef085d671  unannotated.faa
#37ef383804571feb038978980234b231  unannotated.tsv


#Copying the results to Rosalind
scp -r -P 2205 entap_outfiles/ vanessa@143.107.244.181:/home/blowflies/genome
```

Checking md5

```
cd /home/blowflies/genome_annotation/cmeg/11-EnTAP_new/entap_outfiles/final_
md5sum *
#d41d8cd98f00b204e9800998ecf8427e  annotated_contam.faa
#9fc1ae9cc0246a8202519b62f903e047  annotated_contam_gene_ontology_terms.tsv
#12da35de58c8400536adef688be44687  annotated_contam.tsv
#b89d36011f0929530fa3b7b76ce0d1d7  annotated.faa
#0e2f85b9015bb174fd47329979650832  annotated_gene_ontology_terms.tsv
#4438ac6c503707545294f5b43d14cde7  annotated.tsv
#b89d36011f0929530fa3b7b76ce0d1d7  annotated_without_contam.faa
#0e2f85b9015bb174fd47329979650832  annotated_without_contam_gene_ontology_te
#4438ac6c503707545294f5b43d14cde7  annotated_without_contam.tsv
#1a8f2d7eac09b46de2802807c0853d2b  entap_results.tsv
#1e1bd41573cd19d2704e67bef085d671  unannotated.faa
#37ef383804571feb038978980234b231  unannotated.tsv

```

Making an unique gtf file with augustus and ENTAP outputs
files:

- /home/blowflies/genome_annotation/cmeg/11-EnTAP_new/entap_outfiles/final_results/entap_results.tsv
- 1a8f2d7eac09b46de2802807c0853d2b
- /home/diniz/programs/braker/Cmeg_braker_new/second_run/cmeg_braker.gtf
  5b340dc2a5033183337f48b66aac6926
  We downloaded the files to a local computer and checked md5.
  #md5 checked
  In R:

```
# matching AUGUSTUS and ENTAP output into a unique gtf

# libraries
library(data.table)
library(dplyr)

# reading the files
tsv <- fread(file = "entap_results.tsv", header = FALSE)
tsv <- tsv[-1,]
tsv <- tsv[,c(1,13)]
gtf <- fread(file = "braker.gtf")

# updated gtf
new_gtf <- left_join(gtf, tsv, by = c("V9" = "V1"))

# write gtf
fwrite(x = new_gtf, quote = FALSE, sep = '\t', row.names = FALSE,
       col.names = FALSE, file = "cmeg_entap_final.gtf")

# to know how many annotated transcripts are (annoted proteins)
ann_tra <- na.omit(tsv$V13) # look the number of elements in this and compare
```

Total annotated transcripts: 18498 (out of 28019)

We copied the final gtf file to Rosalind server and checked md5:

```
cd /Users/diniz/Desktop
md5 cmeg_entap_final.gtf
#74d1c8756206130b81c9db811ad7713a
scp -P 2205 cmeg_entap_final.gtf diniz@143.107.244.181:/home/blowflies/genome

#On Rosalind:
cd /home/blowflies/genome_annotation/cmeg/12-final_files/
md5sum cmeg_entap_final.gtf
#74d1c8756206130b81c9db811ad7713a
```

12- Final files

```
cd /home/blowflies/genome_annotation/cmeg/12-final_files
md5sum *
#74d1c8756206130b81c9db811ad7713a  Cmeg_annot.gtf
#fc8e26a8c1721f6ba954e0e8c5fdbe9f  Cmeg_cds.fa
#03fccebaf9f74c8e419382978d184655  Cmeg_genome.fa
#410ad1b4fc14eba1faf5e19f2a1951ca  Cmeg_protein.aa
```

type	original/copy	file	Path	md5
Raw genome	original	XDOVE_20221013_S64018_PL100269092-1_B01.ccs.fastq.gz	/home/Reference_genomes/Cmegacephala	4c8b
Raw genome	original	purged.fa	/home/Reference_genomes/Cmegacephala	3224
Raw genome	copy	cmeg.fa	/home/blowflies/genome_annotation/cmeg/0-genome	3224

type	original/copy	file	Path	md5
Mitocondrial genome	original	cmeg_mit.fa	/home/blowflies/genome_annotation/cmeg/0-genome	faff2f427a88295b
Nuclear genome	original	cmeg_N_genome_final.fa	/home/blowflies/genome_annotation/cmeg/0-genome	657b8451d41d97

type	original/copy	file	Path	md5
RNA-seq	original	Cmeg_1.fq.gz	/home/Raw_seqs/cmeg_pool_RNA	9befbbedbbd4e5d9b7
RNA-seq	original	Cmeg_2.fq.gz	/home/Raw_seqs/cmeg_pool_RNA	1be90c92ec0a0e9724
RNA-seq unzipped unzipped	original	Cmeg_1.fq	/home/Raw_seqs/cmeg_pool_RNA	ed5e4a8587d3e36a4a

RNA-seq unzipped	original	Cmeg_2.fq	/home/Raw_seqs/cmeg_pool_RNA	31dbb84405770c7d79
Trimmed reads zipped	original	cmeg_all_1P.fq.gz	/home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all	c21045ff1ef48ace62fd
Trimmed reads zipped	original	cmeg_all_2P.fq.gz	/home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimming/all	8d5c4fe5d822e318dc
Transcriptome	original	cmeg_trinity.fasta	/home/cunha/03-RNA/02-Trinity	ff2f24c065bb88266e7

type	original/copy	file	Path	md5
Masked genome	original	cmeg_N_genome_final.fa.masked	/home/blowflies/genome_annotation/cmeg/5-RepeatMasker	03fccebaf9
Masked genome	old copy	cmeg_masked.fasta	/home/blowflies/genome_annotation/cmeg/9-STAR	a3ead7c04
Masked genome	old copy	Cmeg_masked.fasta	/home/blowflies/genome_annotation/cmeg/10-BRAKER3	a3ead7c04
Masked genome	old copy	Cmeg_masked.fasta	/home/diniz/programs/braker/Cmeg_2/10-BRAKER3	a3ead7c04
Masked genome	copy	Cmeg_masked.fasta	/home/pedro/Non_Coding_Element_Evolution/3-Masking/2-RepeatMasker	03fccebaf9
Masked genome	new copy	cmeg_N_genome_final.fa.masked	/home/diniz/programs/braker/Cmeg_braker_new	03fccebaf9

type	original/copy	file	Path	md5
Proteome	original	cmeg_braker.aa	/home/diniz/programs/braker/Cmeg_braker_new/second_run	410ad1b4fc14
Proteome	copy	cmeg_braker.aa	/home/martins/EnTAP_restart/Proteomes	
Proteome	copy	cmeg_braker.aa	/home/00-Sequences/Cmegacephala/01-Genomic_data/2023	20e2e6100ee
Proteome	copy	cmeg_braker.aa	/home/blowflies/genome_annotation/cmeg/10-Braker_new/Cmeg_braker_new/second_run	410ad1b4fc14

type	original/copy	file	Path	md5
gtf output Braker3 second run	original	cmeg_braker.gtf	/home/diniz/programs/braker/Cmeg_braker_new/second_run	5b340dc2a503
gtf output Braker3 second run	copy	cmeg_braker.gtf	/home/blowflies/genome_annotation/cmeg/10-Braker_new/Cmeg_braker_new/second_run	5b340dc2a503
gtf output Braker3 second run	copy		**Computador do Diniz**	
Final gtf	original		**Computador do Diniz**	
Final gtf	copy		**Rosalind**	

type	original/copy	file	Path	md5

Condingseq	original	braker.codingseq	/home/diniz/programs/braker/Cmeg_braker_new/second_run	fc8e
Condingseq	copy	cmeg_braker.codingseq	/home/blowflies/genome_annotation/cmeg/10-Braker_new/Cmeg_braker_new/second_run	fc8e
Condingseq	copy	cmeg_braker.codingseq	/home/00-Sequences/Cmegacephala/01-Genomic_data/2023	f85

FALTA: conferir md5 dos arquivos usados no script do R do entap e cmeg_entap_final.gtf que está no computador do Diniz

PROBLEMAS:

- **arquivo output do RepeatMasker diferente dos demais que usamos de input nos outros programas. Rodar tudo novamente??**
- **arquivo proteoma da pasta 00-sequences está com md5 diferente do original**
- **arquivo codingseq da pasta 00-sequences está com md5 diferente do original**
- **arquivo gtf que está na pasta 00-sequences não é o final**

OBSOLETE

10- gFACs

https://gfacs.readthedocs.io/en/latest/Flags/index.html

```
cd /home/blowflies/genome_annotation/cmeg/
mkdir 11-gfacs
cd /home/blowflies/genome_annotation/cmeg/11-gfacs
mkdir results
sudo perl /gFACs-master/gFACs.pl -f braker_2.1.2_gtf -p cmeg --rem-all-incom
```

Results:
Number of genes (Augustus/BRAKER): 29938
Number of genes (gFACs): 29258

11- EnTAP

We did it on Darwin

```
cd /home/martins/EnTAP

# The gFACs outputs for all species are here (*_genes.fasta.faa)
mkdir Proteomes

mkdir cmeg
cd cmeg
EnTAP --runP -i /home/martins/EnTAP/Proteomes/cmeg_genes.fasta.faa -d /home/
```

12. gFACs again (with EnTAP output)

```
cd /home/blowflies/genome_annotation/cmeg/
mkdir 13-gfacs_entap
cd /home/blowflies/genome_annotation/cmeg/13-gfacs_entap
mkdir results
sudo perl /gFACs-master/gFACs.pl -f gFACs_gene_table -p cmeg --rem-all-incom
```

13. Final annotation

Final files are here:

```
/dados/home/blowflies/genome_annotation/cmeg/14-final_annot
# 24543 gene models
```

Final busco

```
docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.4.4_cv1 busco -
```

```
***** Results: *****

C:96.5%[S:79.3%,D:17.2%],F:0.9%,M:2.6%,n:3285
3169    Complete BUSCOs (C)
2605    Complete and single-copy BUSCOs (S)
564     Complete and duplicated BUSCOs (D)
29      Fragmented BUSCOs (F)
87      Missing BUSCOs (M)
3285    Total BUSCO groups searched
```

OBSOLETE STUFF

RepeatModeler

```
mkdir 4-RepeatModeler
cp  /home/Reference_genomes/Cmegacephala/purged.fa ./cmeg.fa

#Database
/RepeatModeler-2.0.4/BuildDatabase -name cmeg_database cmeg.fa

#RepeatModeler
screen
/RepeatModeler-2.0.4/RepeatModeler -database cmeg_database -threads 20 -LTRS
#[1] 46887
```

RNA reads from NCBI

We downloded RNA SRAs available from NCBI to use with our own RNA-seq to use it as more evidence for our structural annotation.

We downloaded the SRAs with SRA Toolkit in Darwin. We used a .txt list with the accession numbers.

cmeg_list.txt

SRR1660427
SRR1663113
SRR1663114

They were downloaded in /home/cunha/03-RNA/01-Reads/cmeg/sra_ncbi:

```
prefetch --option-file cmeg_list.txt
```

And then we had to convert the sequences to fastq:

```
fasterq-dump --split-files SRR1660427.sra
fasterq-dump --split-files SRR1663113.sra
fasterq-dump --split-files SRR1663114.sra
```

The SRA files are in /home/cunha/03-RNA/01-Reads/cmeg/sra_ncbi/01-sra_files

Trinity (first)

Transcriptome assembly

Moving trimmed reads to Darwin

```
mkdir /home/cunha/03-RNA/01-Reads/cmeg

scp -P 4988 /home/blowflies/genome_annotation/cmeg/8-cmeg_pool_RNA/2-trimmin
```

Assembling the transcriptome

```
cd /home/cunha/03-RNA/01-Reads/cmeg

docker run --rm -v`pwd`:`pwd` trinityrnaseq/trinityrnaseq Trinity --seqType 

mv cmeg_trinity.* /home/cunha/03-RNA/02-Trinity/cmeg_trinity

# Transcriptome size
grep -c ">" cmeg_trinity.Trinity.fasta # 54372
```

BUSCO (transcriptome quality)

```
mkdir /home/cunha/03-RNA/03-Busco

docker pull ezlabgva/busco:v5.4.4_cv1 # just because we didn't have busco on

# we copied all transcriptomes in this directory and ran everything at once
for i in *; do docker run -u $(id -u) -v $(pwd):/busco_wd ezlabgva/busco:v5.
```

Complete (all)	Complete Single	Complete Dup.	Fragmented	Missing
74.8	36.2	38.6	6.9	18.3

14 - Submission to NCBI

NCBI require a sqn file for assembly submission. I followed the step described in https://www.ncbi.nlm.nih.gov/genbank/genomes_gff/

Then I had to rename the headers in the genome fasta and convert from gtf to gff

```
sed -i '/^>/ s/$/ [organism=Chrysomya megacephala]/' Cmeg_genome.fa
singularity run agat_1.0.0--pl5321hdfd78af_0.sif
agat_convert_sp_gxf2gxf.pl --gtf Cmeg_annot.gtf --output Cmeg_annot.gff
```

FInally I ran table2asn to get the sqn file

```
/home/diniz/programs/linux64.table2asn -M n \
-J \
-c w \
-euk \
-gaps-min 10 \
-f Cmeg_annot.gff \
-i Cmeg_genome.fa \
-locus-tag-prefix Cmeg \
-o cmeg.sqn \
-Z \
-V b
```