Biodiversity Genomics Academy 2023
BUSCO - from QC to gene prediction and phylogenomics
September 20th, 2023

Robert M. Waterhouse, SIB Swiss Institute of Bioinformatics, Switzerland
robert.waterhouse@gmail.com, https://twitter.com/rmwaterhouse, https://ecoevo.social/@rmwaterhouse

Software and Resources for this Practical

These links are provided for information only, they are not needed now for the practical but if you finish any exercises early you may want to investigate the tools/resources a little further by checking out their websites, user guides, and/or their main publications.

OrthoDB - orthology database
Website here; Publication here; Userguide here

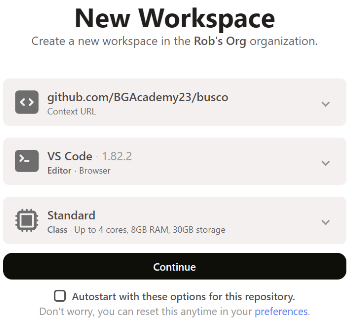
BUSCO - Benchmarking Universal Single-Copy Orthologues
Website here; Publication here; Userguide here

Launching your GitPod Workspace

[1] You must first have a GitHub account
[2] You must first have a GitPod account LINKED to your GitHub account

If you have [1] and [2] then simply clicking this link should launch your Workspace:
https://gitpod.io/#https://github.com/BGAcademy23/busco

Then click Continue with the default settings …



New Workspace

Create a new workspace in the **Rob's Org** organization.

github.com/BGAcademy23/busco
Context URL

VS Code · 1.82.2
Editor · Browser

Standard
Class · Up to 4 cores, 8GB RAM, 30GB storage

☐ Autostart with these options for this repository.
Don't worry, you can reset this anytime in your preferences.

ELSE, see instructions here:
https://docs.google.com/document/d/1gwnq1y80KQ_LwiBJ4mhBfrDfnkqIlWdm8dLj1GQyHQ4/edit?usp=sharing

Exercises: BUSCO - the what, why, and how!

Aim: Use BUSCO to assess the quality of different types of genomic data.

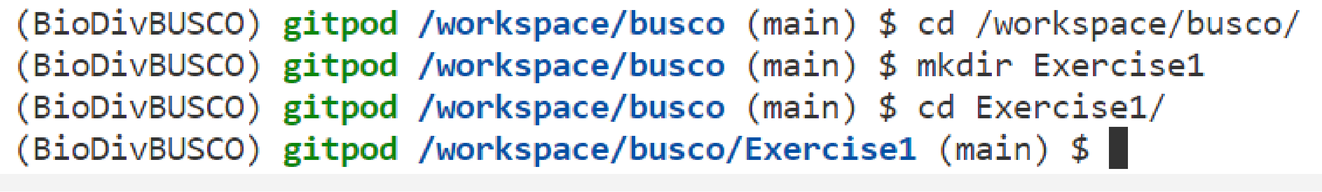
Understanding:

- What are the different BUSCO modes and when should you use them?
- What are the BUSCO outputs and where to find them?

In 2013 I was working on producing genome assemblies for 16 Anopheles mosquitoes, back then we only had short-read technologies and we needed to find a way to assess the quality of the genomes we were producing. I therefore started developing what eventually became BUSCO, now one of the most widely used tools in genomics for assessing the quality of genome assemblies and annotations in terms of gene completeness.

Exercise 1 - Assessing genome assemblies for completeness

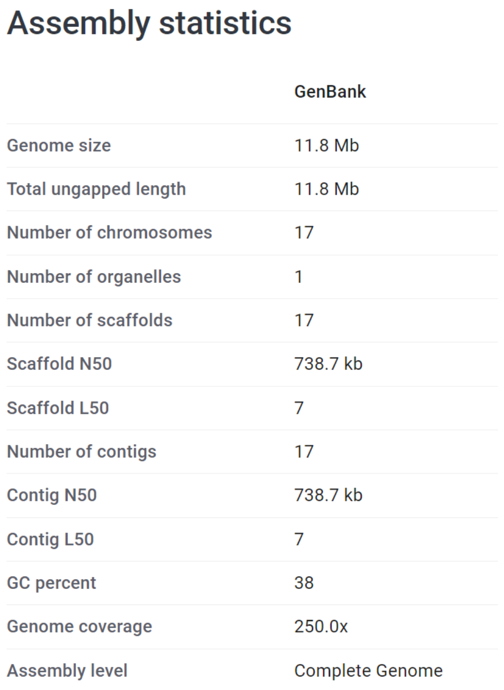
- Before we download any genomics data, let's create a working directory for this exercise, starting by opening a terminal on the Workspace if you've not already got one open. Then, from the **/workspace/busco/** directory ⇒ create a new directory (mkdir) and then navigate into the new directory (cd):
- `cd /workspace/busco/`
- `mkdir Exercise1`
- `cd Exercise1/`



```
(BioDivBUSCO) gitpod /workspace/busco (main) $ cd /workspace/busco/
(BioDivBUSCO) gitpod /workspace/busco (main) $ mkdir Exercise1
(BioDivBUSCO) gitpod /workspace/busco (main) $ cd Exercise1/
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $
```

- Now we can fetch some genome data that we wish to assess
- We will work on a small genome so that it does not take too long to run the analyses, hence I have chosen *Saccharomyces jurei*, a newly discovered fungal species with a small genome of 12 Mbps
  - At NCBI: https://www.ncbi.nlm.nih.gov/datasets/genome/GCA_900290405.1/

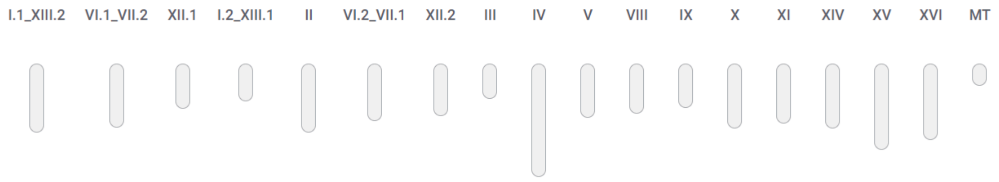
- The summary statistics are already provided by NCBI, but not the BUSCO evaluations



Assembly statistics

	GenBank
Genome size	11.8 Mb
Total ungapped length	11.8 Mb
Number of chromosomes	17
Number of organelles	1
Number of scaffolds	17
Scaffold N50	738.7 kb
Scaffold L50	7
Number of contigs	17
Contig N50	738.7 kb
Contig L50	7
GC percent	38
Genome coverage	250.0x
Assembly level	Complete Genome

Chromosomes

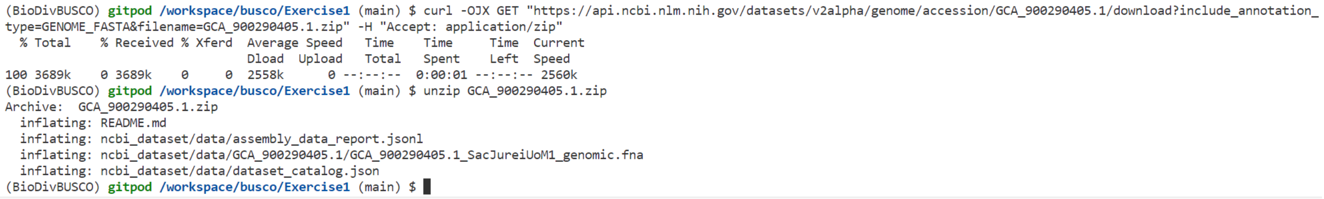


Question: What is the "Scaffold N50" and what does it mean?

Question: What is the "Scaffold L50" and what does it mean?

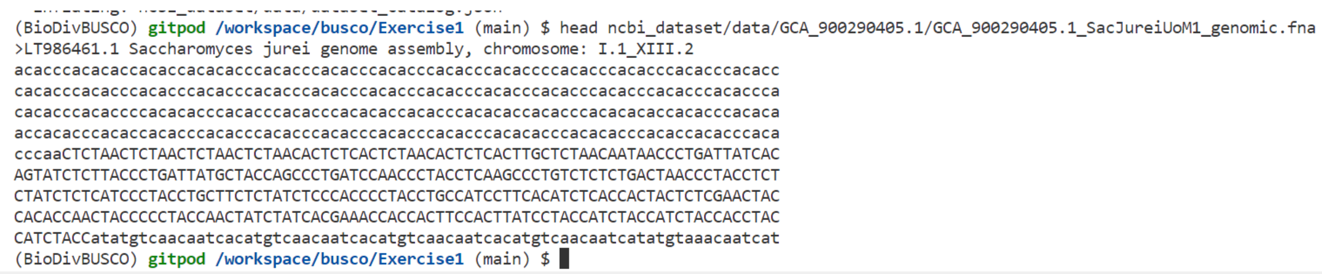
Hints here if needed: https://en.wikipedia.org/wiki/N50,_L50,_and_related_statistics

- We will use the curl command and the NCBI Datasets framework to fetch the genome assembly in FASTA format and then unzip the downloaded file:
- `curl -OJX GET "https://api.ncbi.nlm.nih.gov/datasets/v2alpha/genome/accession/GCA_900290405.1/download?include_annotation_type=GENOME_FASTA&filename=GCA_900290405.1.zip" -H "Accept: application/zip"`
- `unzip GCA_900290405.1.zip`



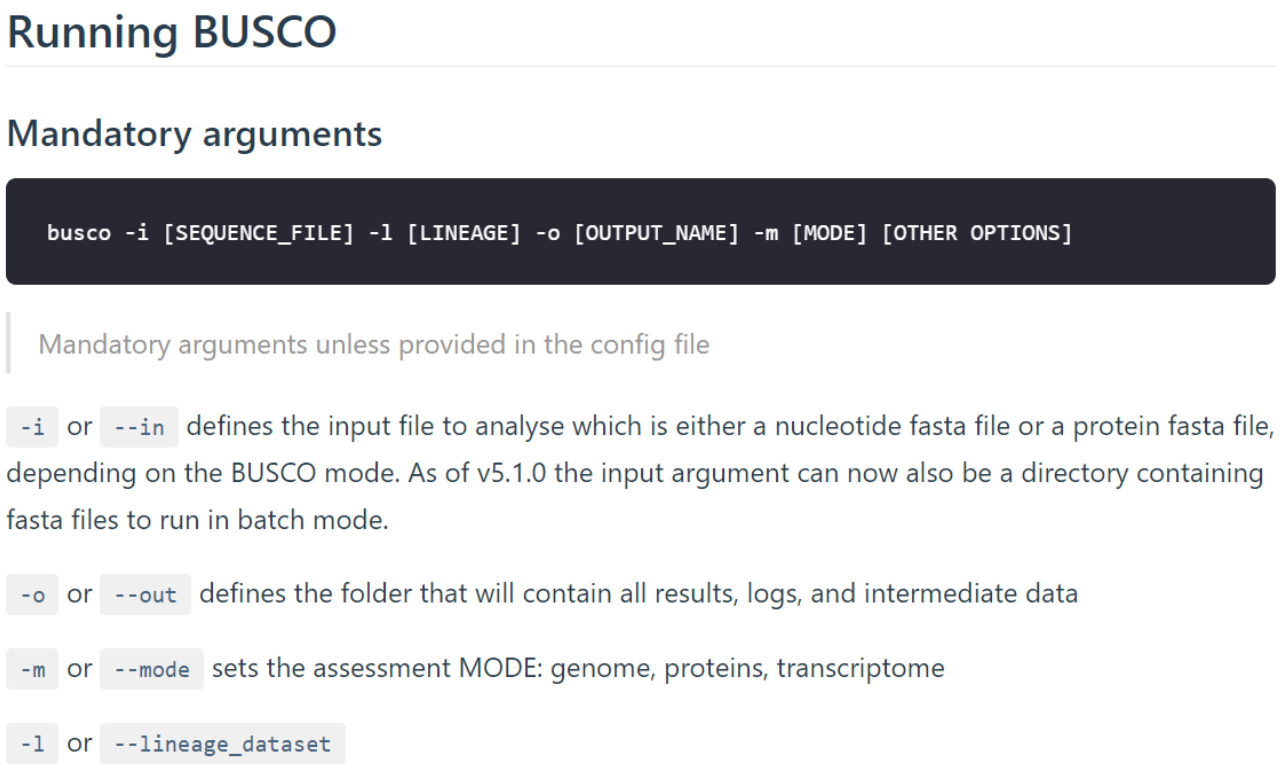
```
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ curl -OJX GET "https://api.ncbi.nlm.nih.gov/datasets/v2alpha/genome/accession/GCA_900290405.1/download?include_annotation_
type=GENOME_FASTA&filename=GCA_900290405.1.zip" -H "Accept: application/zip"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 3689k    0 3689k    0     0  2558k      0 --:--:--  0:00:01 --:--:-- 2560k
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ unzip GCA_900290405.1.zip
Archive:  GCA_900290405.1.zip
  inflating: README.md
  inflating: ncbi_dataset/data/assembly_data_report.jsonl
  inflating: ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna
  inflating: ncbi_dataset/data/dataset_catalog.json
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ █
```

- We will take a look at the first few lines of the FASTA file:
- `head ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna`



```
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ head ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna
>LT986461.1 Saccharomyces jurei genome assembly, chromosome: I.1_XIII.2
acacccacacaccacaccacacacccacacccacacccacacccacacccacaccccacacccacacccacacccacacc
cacacccacacccacacccacacccacacccacacccacacccacacccacacccacacccacacccacacccacaccca
cacacccacaccccacacacccacacccacacccacacaccacacccacacaccacacccacacacaccacacccacaca
accacacccacaccacacccacacccacacccacacccacacccacacccacacacccacacacccacaccacacccaca
cccaaCTCTAACTCTAACTCTAACTCTAACACTCTCACTCTAACACTCTCACTTGCTCTAACAATAACCCTGATTATCAC
AGTATCTCTTACCCTGATTATGCTACCAGCCCTGATCCAACCCTACCTCAAGCCCTGTCTCTCTGACTAACCCTACCTCT
CTATCTCTCATCCCTACCTGCTTCTCTATCTCCCACCCCTACCTGCCATCCTTCACATCTCACCACTACTCTCGAACTAC
CACACCAACTACCCCCTACCAACTATCTATCACGAAACCACCACTTCCACTTATCCTACCATCTACCATCTACCACCTAC
CATCTACCatatgtcaacaatcacatgtcaacaatcacatgtcaacaatcacatgtcaacaatcatatgtaaacaatcat
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $
```

- Now we have the genome, we can run BUSCO assessments in the genome mode to quantify gene completeness of this genome assembly



Running BUSCO

Mandatory arguments

```
busco -i [SEQUENCE_FILE] -l [LINEAGE] -o [OUTPUT_NAME] -m [MODE] [OTHER OPTIONS]
```

> Mandatory arguments unless provided in the config file

`-i` or `--in` defines the input file to analyse which is either a nucleotide fasta file or a protein fasta file, depending on the BUSCO mode. As of v5.1.0 the input argument can now also be a directory containing fasta files to run in batch mode.

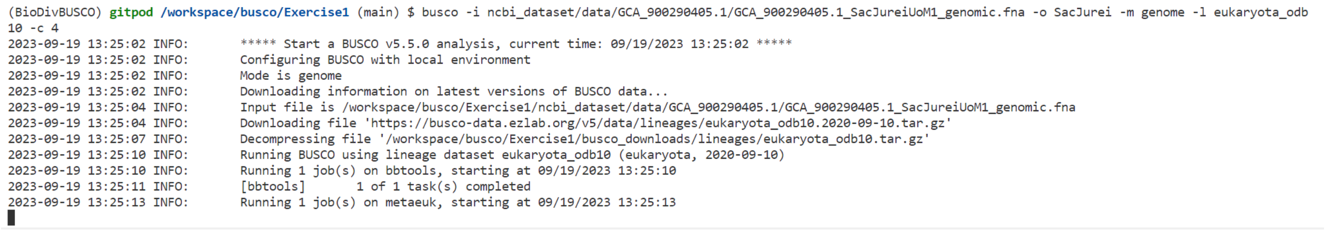
`-o` or `--out` defines the folder that will contain all results, logs, and intermediate data

`-m` or `--mode` sets the assessment MODE: genome, proteins, transcriptome

`-l` or `--lineage_dataset`

- The four main required input options for us therefore are:
  - -i ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna
  - -o SacJurei
  - -m genome
  - -l eukaryota_odb10
- We will also specify the job to use 4 CPUs in order to speed up the task:
  - -c 4
- The whole command will thus be:
- `busco -i ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna -o SacJurei -m genome -l eukaryota_odb10 -c 4`

- On the terminal you can see which steps BUSCO is executing:
    - Configuration
    - Dataset download
    - Metaeuk ⇐ this is the default “gene finding” approach



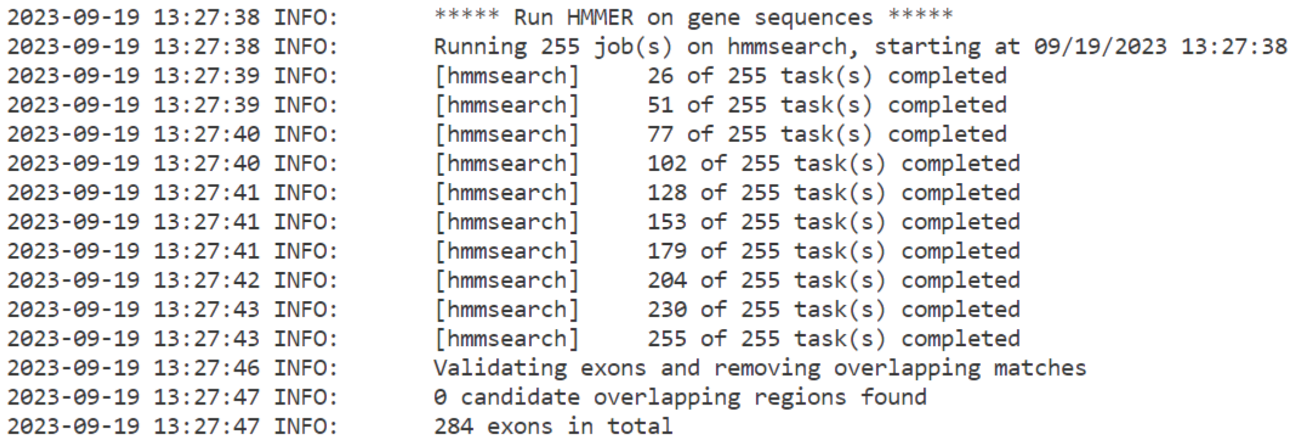
```
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ busco -i ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna -o SacJurei -m genome -l eukaryota_odb
10 -c 4
2023-09-19 13:25:02 INFO:       ***** Start a BUSCO v5.5.0 analysis, current time: 09/19/2023 13:25:02 *****
2023-09-19 13:25:02 INFO:       Configuring BUSCO with local environment
2023-09-19 13:25:02 INFO:       Mode is genome
2023-09-19 13:25:02 INFO:       Downloading information on latest versions of BUSCO data...
2023-09-19 13:25:04 INFO:       Input file is /workspace/busco/Exercise1/ncbi_dataset/data/GCA_900290405.1/GCA_900290405.1_SacJureiUoM1_genomic.fna
2023-09-19 13:25:04 INFO:       Downloading file 'https://busco-data.ezlab.org/v5/data/lineages/eukaryota_odb10.2020-09-10.tar.gz'
2023-09-19 13:25:07 INFO:       Decompressing file '/workspace/busco/Exercise1/busco_downloads/lineages/eukaryota_odb10.tar.gz'
2023-09-19 13:25:10 INFO:       Running BUSCO using lineage dataset eukaryota_odb10 (eukaryota, 2020-09-10)
2023-09-19 13:25:10 INFO:       Running 1 job(s) on bbtools, starting at 09/19/2023 13:25:10
2023-09-19 13:25:11 INFO:       [bbtools]       1 of 1 task(s) completed
2023-09-19 13:25:13 INFO:       Running 1 job(s) on metaeuk, starting at 09/19/2023 13:25:13
█
```

Question: What other “gene finding” approaches are possible to use with BUSCO?

- Which BUSCO lineage to choose - we used the “eukaryota” dataset:
    - eukaryota_odb10 ⇒ 255 BUSCOs
        - fungi_odb10 ⇒ 758 BUSCOs
            - ascomycota_odb10 ⇒ 1706 BUSCOs
                - saccharomycetes_odb10 ⇒ 2137 BUSCOs

Question: Why would you want to use a more specific or a less specific lineage dataset for your BUSCO evaluations?

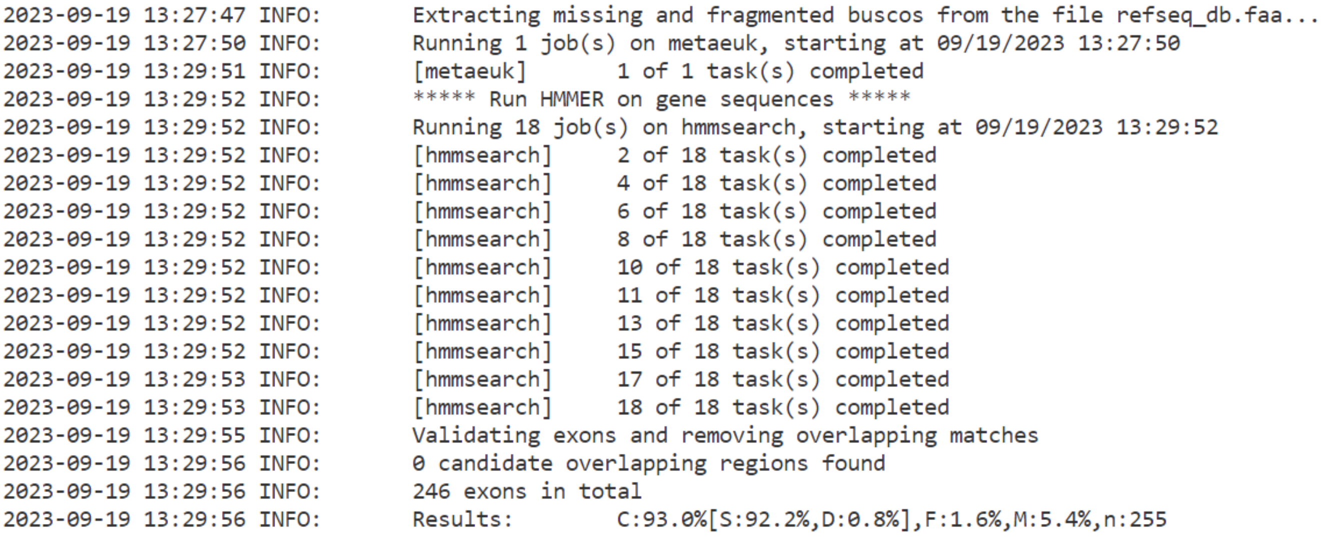
- The analysis continues with the following steps being printed to the terminal:
    - Once metaeuk (first round) is completed, then
    - The hmmsearch step follows



```
2023-09-19 13:27:38 INFO:       ***** Run HMMER on gene sequences *****
2023-09-19 13:27:38 INFO:       Running 255 job(s) on hmmsearch, starting at 09/19/2023 13:27:38
2023-09-19 13:27:39 INFO:       [hmmsearch]     26 of 255 task(s) completed
2023-09-19 13:27:39 INFO:       [hmmsearch]     51 of 255 task(s) completed
2023-09-19 13:27:40 INFO:       [hmmsearch]     77 of 255 task(s) completed
2023-09-19 13:27:40 INFO:       [hmmsearch]     102 of 255 task(s) completed
2023-09-19 13:27:41 INFO:       [hmmsearch]     128 of 255 task(s) completed
2023-09-19 13:27:41 INFO:       [hmmsearch]     153 of 255 task(s) completed
2023-09-19 13:27:41 INFO:       [hmmsearch]     179 of 255 task(s) completed
2023-09-19 13:27:42 INFO:       [hmmsearch]     204 of 255 task(s) completed
2023-09-19 13:27:43 INFO:       [hmmsearch]     230 of 255 task(s) completed
2023-09-19 13:27:43 INFO:       [hmmsearch]     255 of 255 task(s) completed
2023-09-19 13:27:46 INFO:       Validating exons and removing overlapping matches
2023-09-19 13:27:47 INFO:       0 candidate overlapping regions found
2023-09-19 13:27:47 INFO:       284 exons in total
```

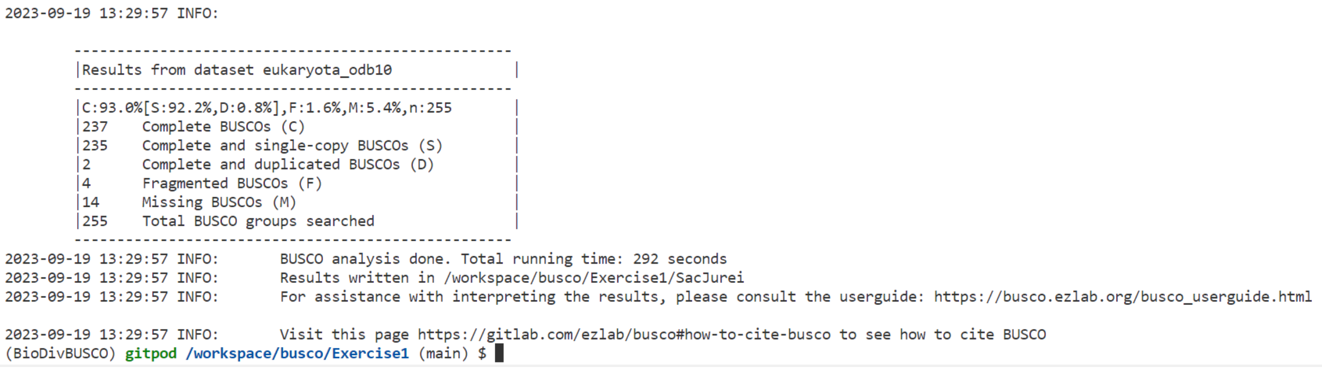
Question: What is the hmmsearch step doing?

- The analysis continues with the following steps being printed to the terminal:
    - The extraction of missing and fragmented buscos
    - A second round of metaeuk predictions
    - Then a second round of hmmsearch follows
    - To finally give the results



```
2023-09-19 13:27:47 INFO:       Extracting missing and fragmented buscos from the file refseq_db.faa...
2023-09-19 13:27:50 INFO:       Running 1 job(s) on metaeuk, starting at 09/19/2023 13:27:50
2023-09-19 13:29:51 INFO:       [metaeuk]       1 of 1 task(s) completed
2023-09-19 13:29:52 INFO:       ***** Run HMMER on gene sequences *****
2023-09-19 13:29:52 INFO:       Running 18 job(s) on hmmsearch, starting at 09/19/2023 13:29:52
2023-09-19 13:29:52 INFO:       [hmmsearch]     2 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     4 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     6 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     8 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     10 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     11 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     13 of 18 task(s) completed
2023-09-19 13:29:52 INFO:       [hmmsearch]     15 of 18 task(s) completed
2023-09-19 13:29:53 INFO:       [hmmsearch]     17 of 18 task(s) completed
2023-09-19 13:29:53 INFO:       [hmmsearch]     18 of 18 task(s) completed
2023-09-19 13:29:55 INFO:       Validating exons and removing overlapping matches
2023-09-19 13:29:56 INFO:       0 candidate overlapping regions found
2023-09-19 13:29:56 INFO:       246 exons in total
2023-09-19 13:29:56 INFO:       Results:        C:93.0%[S:92.2%,D:0.8%],F:1.6%,M:5.4%,n:255
```

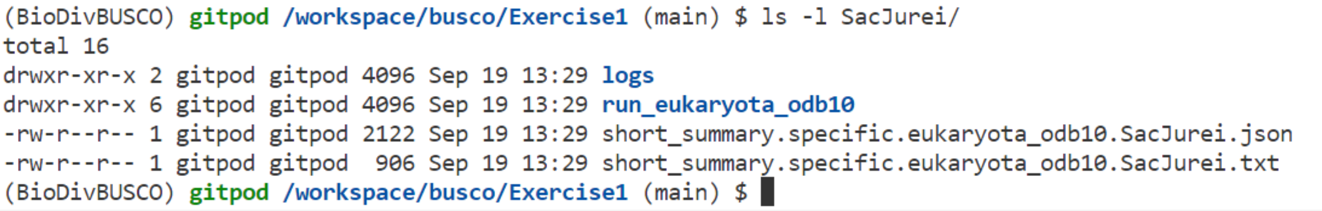
- The analysis should take about 8 minutes to complete



```
2023-09-19 13:29:57 INFO:

        ---------------------------------------------------
        |Results from dataset eukaryota_odb10              |
        ---------------------------------------------------
        |C:93.0%[S:92.2%,D:0.8%],F:1.6%,M:5.4%,n:255       |
        |237    Complete BUSCOs (C)                        |
        |235    Complete and single-copy BUSCOs (S)        |
        |2      Complete and duplicated BUSCOs (D)         |
        |4      Fragmented BUSCOs (F)                      |
        |14     Missing BUSCOs (M)                         |
        |255    Total BUSCO groups searched                |
        ---------------------------------------------------
2023-09-19 13:29:57 INFO:       BUSCO analysis done. Total running time: 292 seconds
2023-09-19 13:29:57 INFO:       Results written in /workspace/busco/Exercise1/SacJurei
2023-09-19 13:29:57 INFO:       For assistance with interpreting the results, please consult the userguide: https://busco.ezlab.org/busco_userguide.html

2023-09-19 13:29:57 INFO:       Visit this page https://gitlab.com/ezlab/busco#how-to-cite-busco to see how to cite BUSCO
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ █
```

- If BUSCO failed, you can get the file from here instead, using ‘wget’:
- `wget https://apollo.vital-it.ch/trackvis/BGA23/SacJurei.zip`
- `unzip SacJurei.zip`

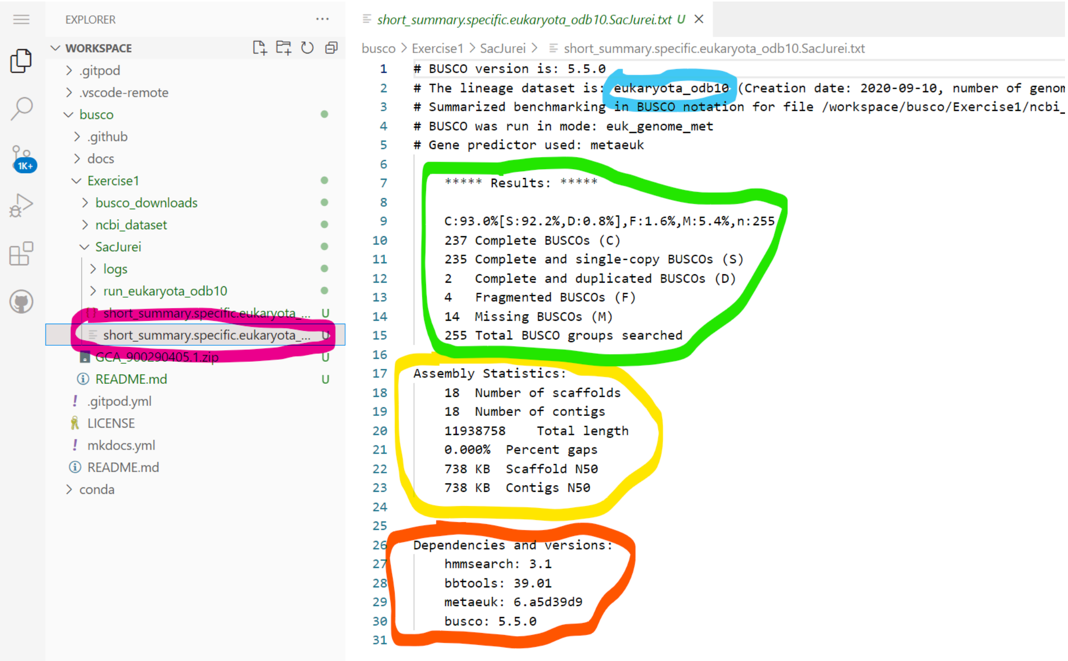
- Let’s explore the results of a typical genome assembly assessment run:
- `ls -l SacJurei/`



```
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ ls -l SacJurei/
total 16
drwxr-xr-x 2 gitpod gitpod 4096 Sep 19 13:29 logs
drwxr-xr-x 6 gitpod gitpod 4096 Sep 19 13:29 run_eukaryota_odb10
-rw-r--r-- 1 gitpod gitpod 2122 Sep 19 13:29 short_summary.specific.eukaryota_odb10.SacJurei.json
-rw-r--r-- 1 gitpod gitpod  906 Sep 19 13:29 short_summary.specific.eukaryota_odb10.SacJurei.txt
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ █
```

- **logs** - useful if something went wrong
- **short_summary.specific.eukaryota_odb10.SacJurei.json** (JSON version)

- **short_summary.specific.eukaryota_odb10.SacJurei.txt** - open the file in the text editor (pink, explorer)
    - Indicates the lineage dataset that was used (blue)
    - Summarises the main results (green)
    - Provides some assembly statistics (yellow)
    - Lists the versions of all the tools used during this run (orange)



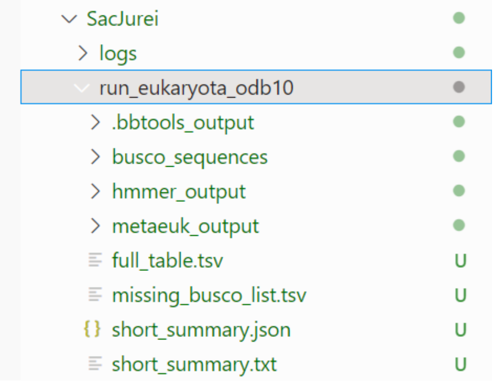
```
EXPLORER ...
WORKSPACE
 > .gitpod
 > .vscode-remote
 v busco
   > .github
   > docs
   v Exercise1
     > busco_downloads
     > ncbi_dataset
     v SacJurei
       > logs
       > run_eukaryota_odb10
       short_summary.specific.eukaryota ...   U
       short_summary.specific.eukaryota_...   U
     GCA_900290405.1.zip                      U
     README.md                                U
   .gitpod.yml
   LICENSE
   mkdocs.yml
   README.md
 > conda
```

short_summary.specific.eukaryota_odb10.SacJurei.txt U

busco > Exercise1 > SacJurei > short_summary.specific.eukaryota_odb10.SacJurei.txt

```
# BUSCO version is: 5.5.0
# The lineage dataset is: eukaryota_odb10 (Creation date: 2020-09-10, number of genom
# Summarized benchmarking in BUSCO notation for file /workspace/busco/Exercise1/ncbi_
# BUSCO was run in mode: euk_genome_met
# Gene predictor used: metaeuk

    ***** Results: *****

    C:93.0%[S:92.2%,D:0.8%],F:1.6%,M:5.4%,n:255
    237 Complete BUSCOs (C)
    235 Complete and single-copy BUSCOs (S)
    2   Complete and duplicated BUSCOs (D)
    4   Fragmented BUSCOs (F)
    14  Missing BUSCOs (M)
    255 Total BUSCO groups searched

Assembly Statistics:
    18  Number of scaffolds
    18  Number of contigs
    11938758    Total length
    0.000%  Percent gaps
    738 KB  Scaffold N50
    738 KB  Contigs N50


Dependencies and versions:
    hmmsearch: 3.1
    bbtools: 39.01
    metaeuk: 6.a5d39d9
    busco: 5.5.0
```

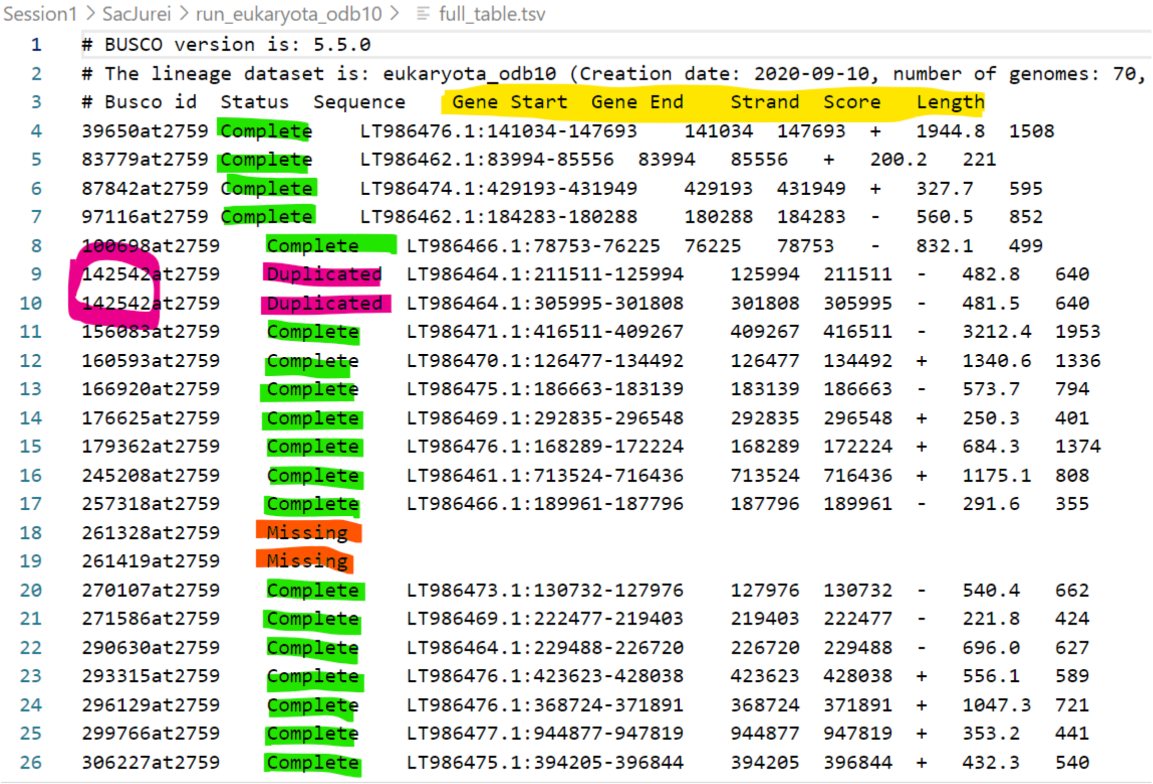
- **run_eukaryota_odb10** - folder with the full results from the run



```
v SacJurei
  > logs
  v run_eukaryota_odb10
    > .bbtools_output
    > busco_sequences
    > hmmer_output
    > metaeuk_output
    full_table.tsv              U
    missing_busco_list.tsv      U
    short_summary.json          U
    short_summary.txt           U
```

    - **busco_sequences**: PROTEIN and DNA sequences provided
        - **fragmented_busco_sequences**
        - **multi_copy_busco_sequences**
        - **single_copy_busco_sequences**

- **full_table.tsv**
    - Details of status (Complete, Duplicated, Fragmented, or Missing), genomic locations, scores, and lengths of all searched BUSCOs



```
# BUSCO version is: 5.5.0
# The lineage dataset is: eukaryota_odb10 (Creation date: 2020-09-10, number of genomes: 70,
# Busco id  Status  Sequence    Gene Start  Gene End    Strand  Score   Length
39650at2759 Complete    LT986476.1:141034-147693    141034  147693  +   1944.8  1508
83779at2759 Complete    LT986462.1:83994-85556  83994   85556   +   200.2   221
87842at2759 Complete    LT986474.1:429193-431949    429193  431949  +   327.7   595
97116at2759 Complete    LT986462.1:184283-180288    180288  184283  -   560.5   852
100698at2759    Complete    LT986466.1:78753-76225  76225   78753   -   832.1   499
142542at2759    Duplicated  LT986464.1:211511-125994    125994  211511  -   482.8   640
142542at2759    Duplicated  LT986464.1:305995-301808    301808  305995  -   481.5   640
156083at2759    Complete    LT986471.1:416511-409267    409267  416511  -   3212.4  1953
160593at2759    Complete    LT986470.1:126477-134492    126477  134492  +   1340.6  1336
166920at2759    Complete    LT986475.1:186663-183139    183139  186663  -   573.7   794
176625at2759    Complete    LT986469.1:292835-296548    292835  296548  +   250.3   401
179362at2759    Complete    LT986476.1:168289-172224    168289  172224  +   684.3   1374
245208at2759    Complete    LT986461.1:713524-716436    713524  716436  +   1175.1  808
257318at2759    Complete    LT986466.1:189961-187796    187796  189961  -   291.6   355
261328at2759    Missing
261419at2759    Missing
270107at2759    Complete    LT986473.1:130732-127976    127976  130732  -   540.4   662
271586at2759    Complete    LT986469.1:222477-219403    219403  222477  -   221.8   424
290630at2759    Complete    LT986464.1:229488-226720    226720  229488  -   696.0   627
293315at2759    Complete    LT986476.1:423623-428038    423623  428038  +   556.1   589
296129at2759    Complete    LT986476.1:368724-371891    368724  371891  +   1047.3  721
299766at2759    Complete    LT986477.1:944877-947819    944877  947819  +   353.2   441
306227at2759    Complete    LT986475.1:394205-396844    394205  396844  +   432.3   540
```

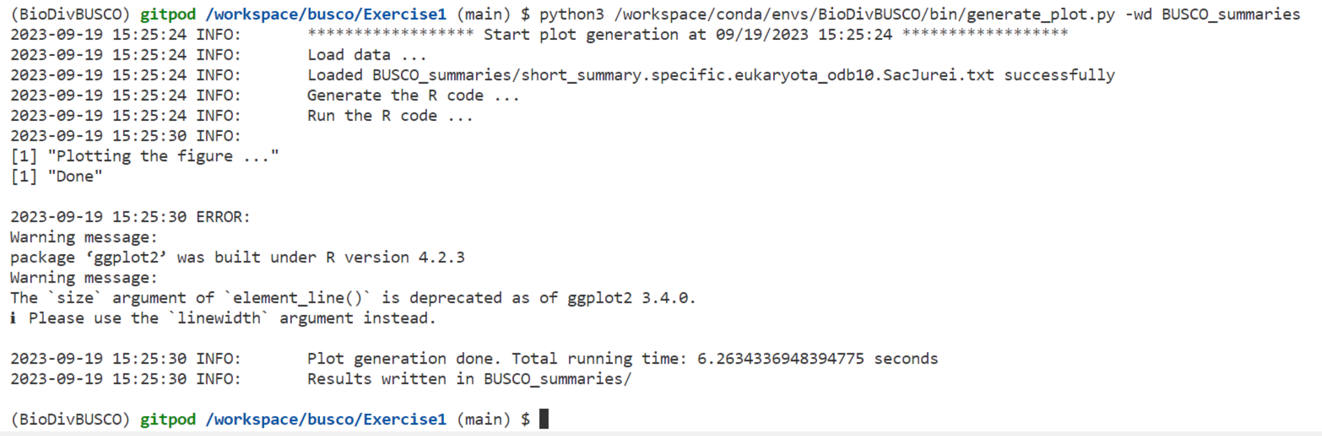
- **hmmer_output** - searching predicted proteins against BUSCO profiles
    - initial_run_results
    - Rerun_results
- **metaeuk_output** - the gene prediction results
    - combined_nucl_seqs.fas
    - combined_pred_proteins.fas
    - initial_results
    - refseq_db_rerun.faa
    - Rerun_results
- **missing_busco_list.tsv**
    - The BUSCOs that were never found

- Let's plot the results of a typical genome assembly assessment run. BUSCO provides R scripts to produce basic plots that summarise the results
- Normally you would want to summarise the results from several assessments, e.g. the same genome with different lineage datasets, or different versions of your genome assembly build with different workflows/parameters, or the assessments of the genome assemblies of several species side-by-side
- Therefore we need to copy our short summary file into a new folder

To run `scripts/generate_plot.py`, first create a folder, e.g. `mkdir BUSCO_summaries`, and then copy the BUSCO short summary file from each of the runs you want to plot into this folder.

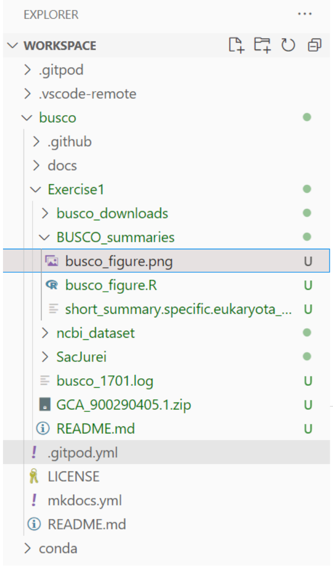
- `mkdir BUSCO_summaries`
- `cp SacJurei/short_summary.specific.eukaryota_odb10.SacJurei.txt BUSCO_summaries/`
    - Make sure you are still in the Exercise1 directory!
    - You can use the command `pwd` to check where you are located

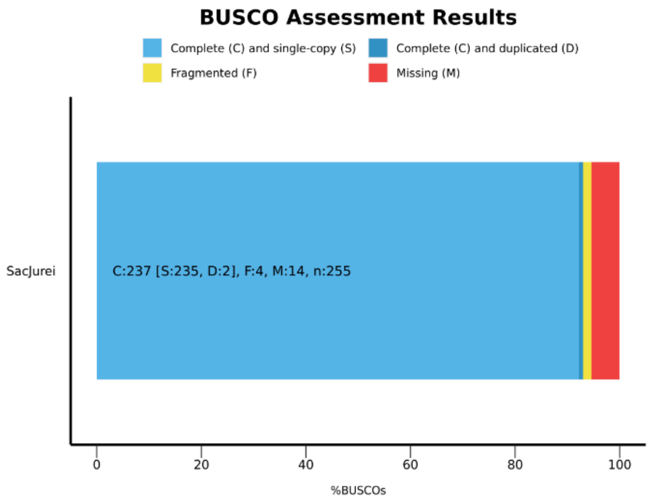
- Run the python script to plot the summary:
- `python3 /workspace/conda/envs/BioDivBUSCO/bin/generate_plot.py -wd BUSCO_summaries`
    - This command will produce some warnings and errors but you can ignore them - they are caused by an argument deprecation in the R package ggplot2



```
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ python3 /workspace/conda/envs/BioDivBUSCO/bin/generate_plot.py -wd BUSCO_summaries
2023-09-19 15:25:24 INFO:       ****************** Start plot generation at 09/19/2023 15:25:24 ******************
2023-09-19 15:25:24 INFO:       Load data ...
2023-09-19 15:25:24 INFO:       Loaded BUSCO_summaries/short_summary.specific.eukaryota_odb10.SacJurei.txt successfully
2023-09-19 15:25:24 INFO:       Generate the R code ...
2023-09-19 15:25:24 INFO:       Run the R code ...
2023-09-19 15:25:30 INFO:
[1] "Plotting the figure ..."
[1] "Done"

2023-09-19 15:25:30 ERROR:
Warning message:
package ‘ggplot2’ was built under R version 4.2.3
Warning message:
The `size` argument of `element_line()` is deprecated as of ggplot2 3.4.0.
ℹ Please use the `linewidth` argument instead.

2023-09-19 15:25:30 INFO:       Plot generation done. Total running time: 6.2634336948394775 seconds
2023-09-19 15:25:30 INFO:       Results written in BUSCO_summaries/

(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ █
```

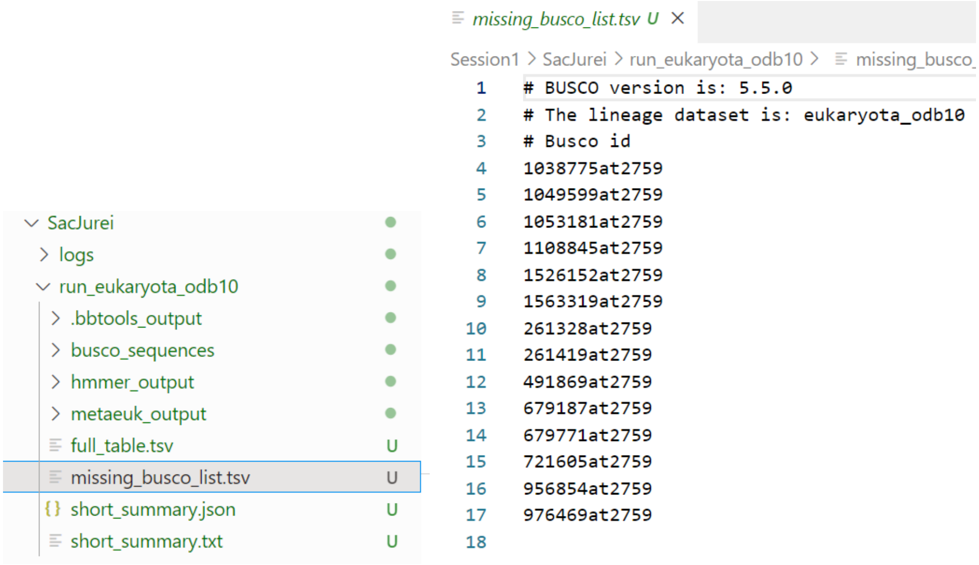
- Open the resulting plot file (busco_figure.png) using the explorer (left panel of the window) - you can zoom in/out by using Ctrl + scroll up/down (mousewheel) on the figure (if it did not work for you, see the plot here)





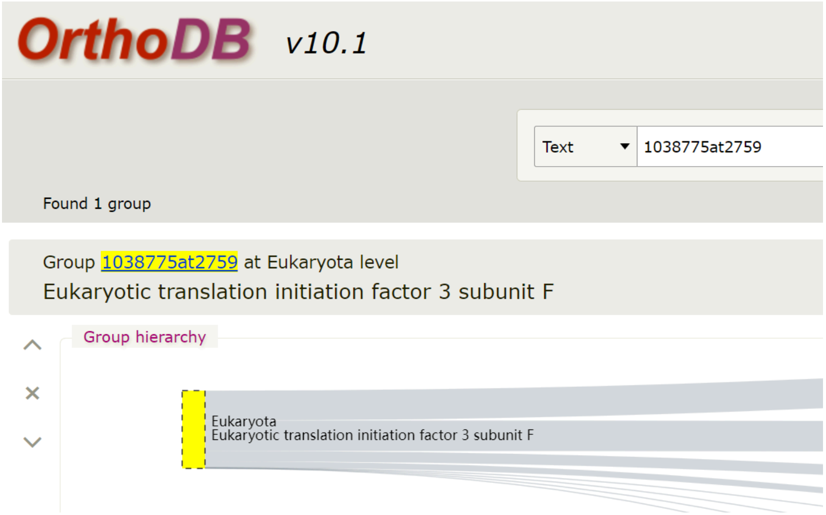
Question: What does the red part of the plot show and how many BUSCOs does it indicate?

- Let’s check some of these apparently missing BUSCOs



```
# BUSCO version is: 5.5.0
# The lineage dataset is: eukaryota_odb10
# Busco id
1038775at2759
1049599at2759
1053181at2759
1108845at2759
1526152at2759
1563319at2759
261328at2759
261419at2759
491869at2759
679187at2759
679771at2759
721605at2759
956854at2759
976469at2759

```

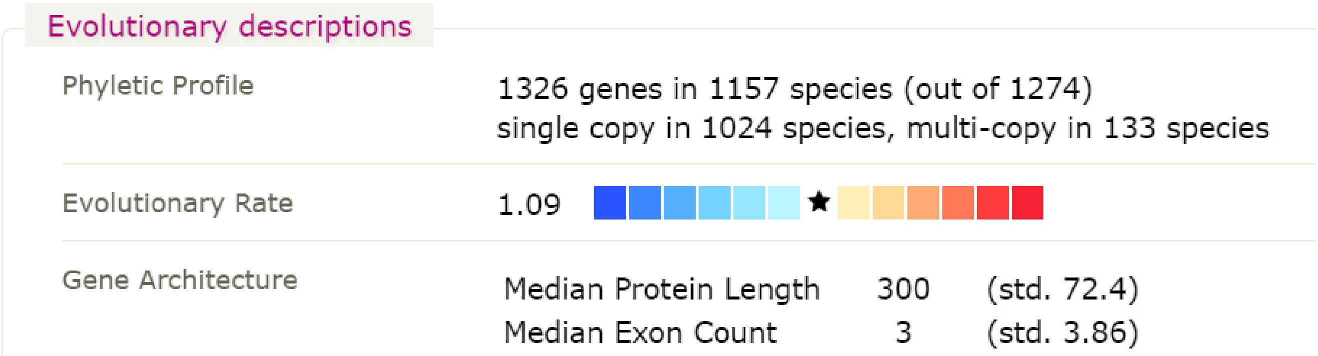
- Search OrthoDB v10 for the missing BUSCO, [1038775at2759]
- Eukaryotic translation initiation factor 3 subunit F



Found 1 group

Group 1038775at2759 at Eukaryota level
Eukaryotic translation initiation factor 3 subunit F

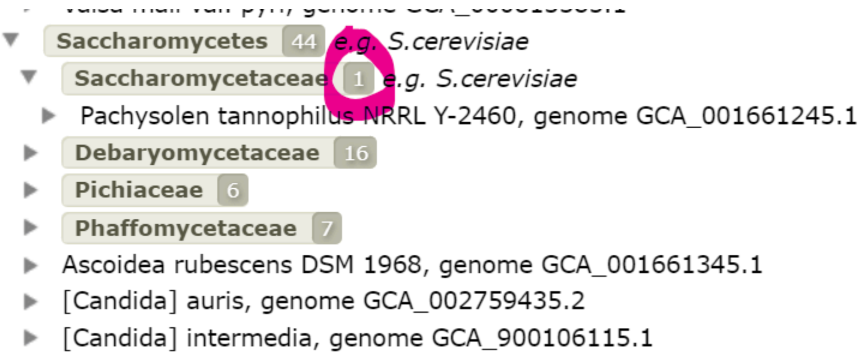
- This gene has orthologues in 91% of eukaryotes at OrthoDB v10
- Of these, it is single-copy in 89%



Evolutionary descriptions

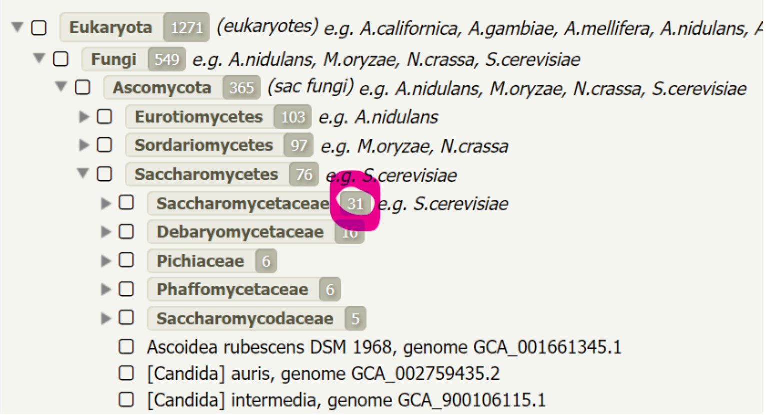
Phyletic Profile	1326 genes in 1157 species (out of 1274) single copy in 1024 species, multi-copy in 133 species
Evolutionary Rate	1.09 🟦🟦🟦🟦🟦🟦 ★ 🟨🟨🟧🟧🟥🟥
Gene Architecture	Median Protein Length 300 (std. 72.4) Median Exon Count 3 (std. 3.86)

- Checking other Saccharomyces species at OrthoDB in the same orthogroup, 1038775at2759, reveals that only one species of Saccharomycetaceae seems to have an orthologue of Eukaryotic translation initiation factor 3 subunit F



▼ Saccharomycetes 44 e.g. S.cerevisiae
▼ Saccharomycetaceae 1 e.g. S.cerevisiae
▶ Pachysolen tannophilus NRRL Y-2460, genome GCA_001661245.1
▶ Debaryomycetaceae 16
▶ Pichiaceae 6
▶ Phaffomycetaceae 7
▶ Ascoidea rubescens DSM 1968, genome GCA_001661345.1
▶ [Candida] auris, genome GCA_002759435.2
▶ [Candida] intermedia, genome GCA_900106115.1

- This is despite the fact that OrthoDB v10 contains 31 species of Saccharomycetaceae, strongly suggesting a real evolutionary loss of this otherwise highly conserved gene



▼ □ Eukaryota 1271 (eukaryotes) e.g. A.californica, A.gambiae, A.mellifera, A.nidulans, A
▼ □ Fungi 549 e.g. A.nidulans, M.oryzae, N.crassa, S.cerevisiae
▼ □ Ascomycota 365 (sac fungi) e.g. A.nidulans, M.oryzae, N.crassa, S.cerevisiae
▶ □ Eurotiomycetes 103 e.g. A.nidulans
▶ □ Sordariomycetes 97 e.g. M.oryzae, N.crassa
▼ □ Saccharomycetes 76 e.g. S.cerevisiae
▶ □ Saccharomycetaceae 31 e.g. S.cerevisiae
▶ □ Debaryomycetaceae 16
▶ □ Pichiaceae 6
▶ □ Phaffomycetaceae 6
▶ □ Saccharomycodaceae 5
□ Ascoidea rubescens DSM 1968, genome GCA_001661345.1
□ [Candida] auris, genome GCA_002759435.2
□ [Candida] intermedia, genome GCA_900106115.1

- This has been recognised in the literature, and indeed two of the other missing BUSCOs are the group of Subunit H orthologues, 976469at2759, and the group of Subunit M orthologues, 679771at2759

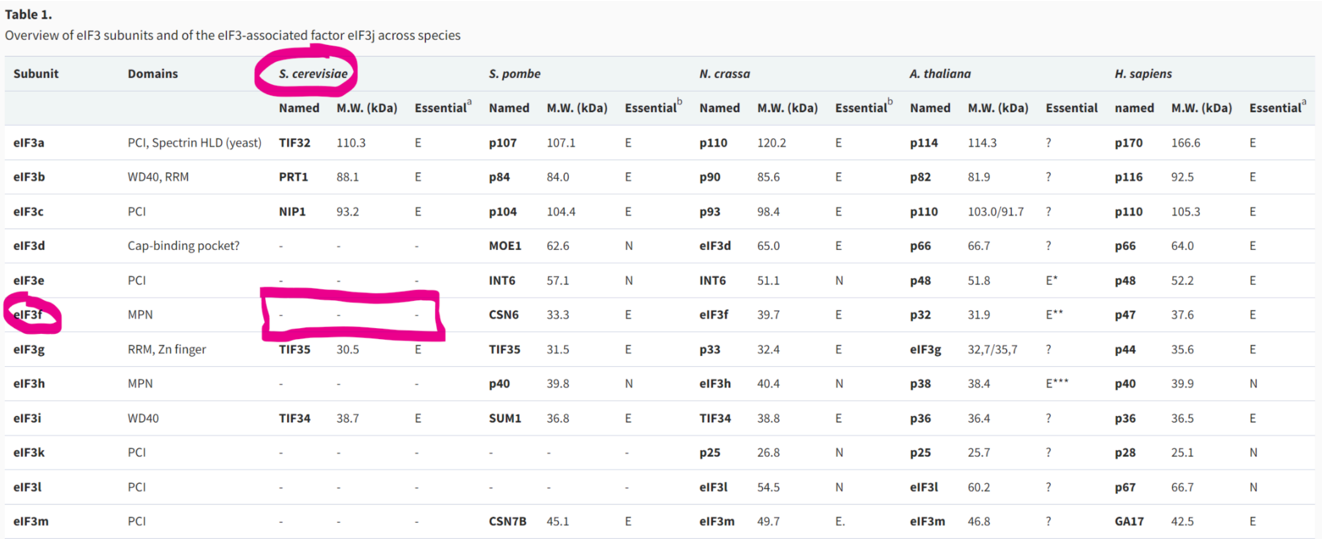


Table 1.
Overview of eIF3 subunits and of the eIF3-associated factor eIF3j across species

Subunit	Domains	*S. cerevisiae*			*S. pombe*			*N. crassa*			*A. thaliana*			*H. sapiens*		
		Named	M.W. (kDa)	Essential[a]	Named	M.W. (kDa)	Essential[b]	Named	M.W. (kDa)	Essential[b]	Named	M.W. (kDa)	Essential	named	M.W. (kDa)	Essential[a]
eIF3a	PCI, Spectrin HLD (yeast)	**TIF32**	110.3	E	**p107**	107.1	E	**p110**	120.2	E	**p114**	114.3	?	**p170**	166.6	E
eIF3b	WD40, RRM	**PRT1**	88.1	E	**p84**	84.0	E	**p90**	85.6	E	**p82**	81.9	?	**p116**	92.5	E
eIF3c	PCI	**NIP1**	93.2	E	**p104**	104.4	E	**p93**	98.4	E	**p110**	103.0/91.7	?	**p110**	105.3	E
eIF3d	Cap-binding pocket?	-	-	-	**MOE1**	62.6	N	**eIF3d**	65.0	E	**p66**	66.7	?	**p66**	64.0	E
eIF3e	PCI	-	-	-	**INT6**	57.1	N	**INT6**	51.1	N	**p48**	51.8	E*	**p48**	52.2	E
eIF3f	MPN	-	-	-	**CSN6**	33.3	E	**eIF3f**	39.7	E	**p32**	31.9	E**	**p47**	37.6	E
eIF3g	RRM, Zn finger	**TIF35**	30.5	E	**TIF35**	31.5	E	**p33**	32.4	E	**eIF3g**	32,7/35,7	?	**p44**	35.6	E
eIF3h	MPN	-	-	-	**p40**	39.8	N	**eIF3h**	40.4	N	**p38**	38.4	E***	**p40**	39.9	N
eIF3i	WD40	**TIF34**	38.7	E	**SUM1**	36.8	E	**TIF34**	38.8	E	**p36**	36.4	?	**p36**	36.5	E
eIF3k	PCI	-	-	-	-	-	-	**p25**	26.8	N	**p25**	25.7	?	**p28**	25.1	N
eIF3l	PCI	-	-	-	-	-	-	**eIF3l**	54.5	N	**eIF3l**	60.2	?	**p67**	66.7	N
eIF3m	PCI	-	-	-	**CSN7B**	45.1	E	**eIF3m**	49.7	E.	**eIF3m**	46.8	?	**GA17**	42.5	E

Question: What can you conclude from this investigation?

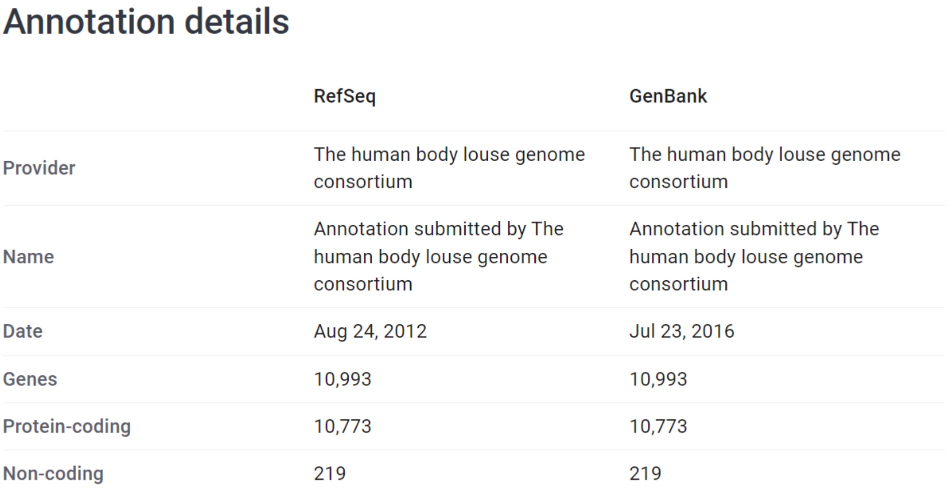
Exercise 2 - Assessing genome annotations for completeness

Next we will run an assessment of an annotated protein set, *i.e.* “proteins” mode.

Question: What is the difference between “proteins” mode and “transcriptome” mode?

- Before we download any genomics data, let’s create a working directory for this exercise, starting by opening a terminal on the Workspace if you’ve not already got one open. Then, from the **/workspace/busco/** directory ⇒ create a new directory (mkdir) and then navigate into the new directory (cd):
- `cd /workspace/busco/`
- `mkdir Exercise2`
- `cd Exercise2/`

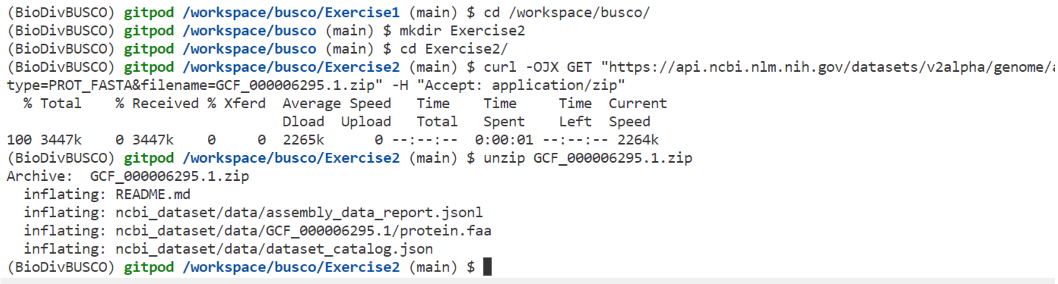
- For this exercise we will look at the human body louse, *Pediculus humanus corporis*
  - At NCBI: https://www.ncbi.nlm.nih.gov/datasets/genome/GCF_000006295.1/
- The NCBI reports a total of 10’773 protein-coding genes, which is a somewhat small number of genes compared to many other insect genomes



Annotation details

	RefSeq	GenBank
Provider	The human body louse genome consortium	The human body louse genome consortium
Name	Annotation submitted by The human body louse genome consortium	Annotation submitted by The human body louse genome consortium
Date	Aug 24, 2012	Jul 23, 2016
Genes	10,993	10,993
Protein-coding	10,773	10,773
Non-coding	219	219

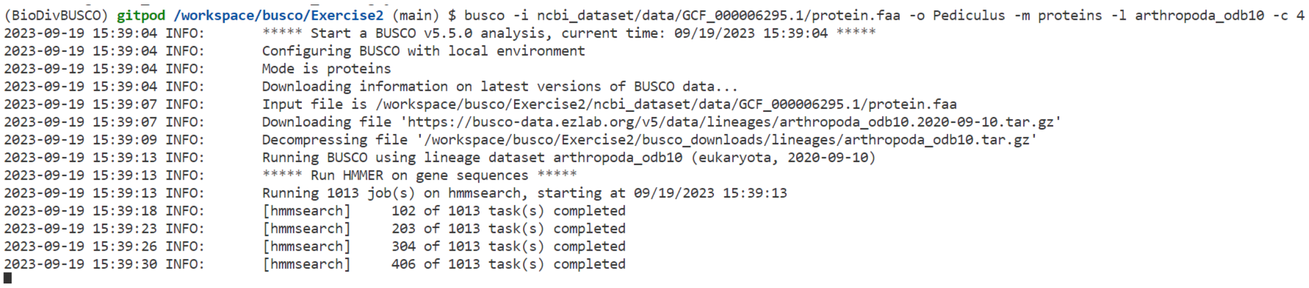
- We will use the curl command and the NCBI Datasets framework to fetch the genome annotations file in FASTA format and then unzip the downloaded file:
- `curl -OJX GET "https://api.ncbi.nlm.nih.gov/datasets/v2alpha/genome/accession/GCF_000006295.1/download?include_annotation_type=PROT_FASTA&filename=GCF_000006295.1.zip" -H "Accept: application/zip"`
- `unzip GCF_000006295.1.zip`



```
(BioDivBUSCO) gitpod /workspace/busco/Exercise1 (main) $ cd /workspace/busco/
(BioDivBUSCO) gitpod /workspace/busco (main) $ mkdir Exercise2
(BioDivBUSCO) gitpod /workspace/busco (main) $ cd Exercise2/
(BioDivBUSCO) gitpod /workspace/busco/Exercise2 (main) $ curl -OJX GET "https://api.ncbi.nlm.nih.gov/datasets/v2alpha/genome/a
type=PROT_FASTA&filename=GCF_000006295.1.zip" -H "Accept: application/zip"
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 3447k    0 3447k    0     0  2265k      0 --:--:--  0:00:01 --:--:-- 2264k
(BioDivBUSCO) gitpod /workspace/busco/Exercise2 (main) $ unzip GCF_000006295.1.zip
Archive:  GCF_000006295.1.zip
  inflating: README.md
  inflating: ncbi_dataset/data/assembly_data_report.jsonl
  inflating: ncbi_dataset/data/GCF_000006295.1/protein.faa
  inflating: ncbi_dataset/data/dataset_catalog.json
(BioDivBUSCO) gitpod /workspace/busco/Exercise2 (main) $ 
```

- The four main required input options for us therefore are:
    - -i ncbi_dataset/data/GCF_000006295.1/protein.faa
    - -o Pediculus
    - -m proteins
    - -l arthropoda_odb10
- We will also specify the job to use 4 CPUs in order to speed up the task
    - -c 4
- The whole command will thus be:
- `busco -i ncbi_dataset/data/GCF_000006295.1/protein.faa -o Pediculus -m proteins -l arthropoda_odb10 -c 4`

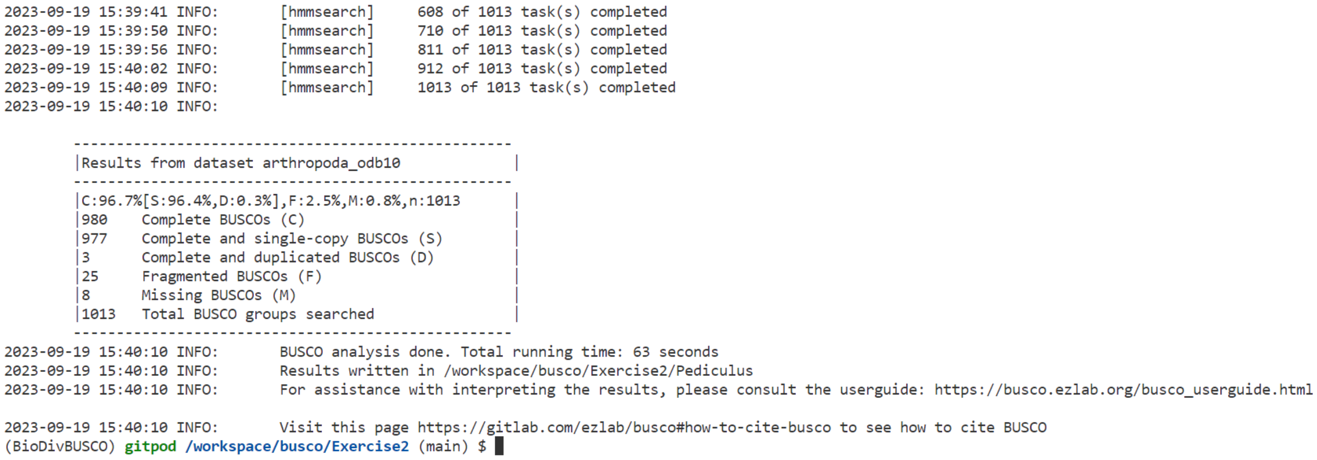
- The job starts by configuring the environment, then fetching the "arthropoda_odb10" lineage dataset, and then launching the hmmsearch jobs



```
(BioDivBUSCO) gitpod /workspace/busco/Exercise2 (main) $ busco -i ncbi_dataset/data/GCF_000006295.1/protein.faa -o Pediculus -m proteins -l arthropoda_odb10 -c 4
2023-09-19 15:39:04 INFO:       ***** Start a BUSCO v5.5.0 analysis, current time: 09/19/2023 15:39:04 *****
2023-09-19 15:39:04 INFO:       Configuring BUSCO with local environment
2023-09-19 15:39:04 INFO:       Mode is proteins
2023-09-19 15:39:04 INFO:       Downloading information on latest versions of BUSCO data...
2023-09-19 15:39:07 INFO:       Input file is /workspace/busco/Exercise2/ncbi_dataset/data/GCF_000006295.1/protein.faa
2023-09-19 15:39:07 INFO:       Downloading file 'https://busco-data.ezlab.org/v5/data/lineages/arthropoda_odb10.2020-09-10.tar.gz'
2023-09-19 15:39:09 INFO:       Decompressing file '/workspace/busco/Exercise2/busco_downloads/lineages/arthropoda_odb10.tar.gz'
2023-09-19 15:39:13 INFO:       Running BUSCO using lineage dataset arthropoda_odb10 (eukaryota, 2020-09-10)
2023-09-19 15:39:13 INFO:       ***** Run HMMER on gene sequences *****
2023-09-19 15:39:13 INFO:       Running 1013 job(s) on hmmsearch, starting at 09/19/2023 15:39:13
2023-09-19 15:39:18 INFO:       [hmmsearch]     102 of 1013 task(s) completed
2023-09-19 15:39:23 INFO:       [hmmsearch]     203 of 1013 task(s) completed
2023-09-19 15:39:26 INFO:       [hmmsearch]     304 of 1013 task(s) completed
2023-09-19 15:39:30 INFO:       [hmmsearch]     406 of 1013 task(s) completed
```

Question: Why does BUSCO this time go directly into the hmmsearch steps and not start with the gene prediction (metaeuk) steps?

- When the 1013 searches are complete, the final results are displayed

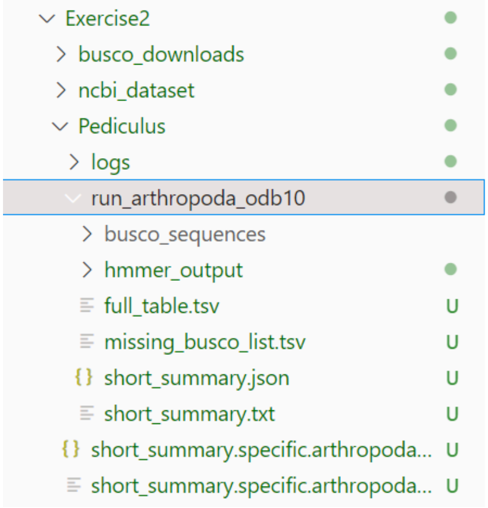


```
2023-09-19 15:39:41 INFO:       [hmmsearch]     608 of 1013 task(s) completed
2023-09-19 15:39:50 INFO:       [hmmsearch]     710 of 1013 task(s) completed
2023-09-19 15:39:56 INFO:       [hmmsearch]     811 of 1013 task(s) completed
2023-09-19 15:40:02 INFO:       [hmmsearch]     912 of 1013 task(s) completed
2023-09-19 15:40:09 INFO:       [hmmsearch]     1013 of 1013 task(s) completed
2023-09-19 15:40:10 INFO:

        ---------------------------------------------------
        |Results from dataset arthropoda_odb10             |
        ---------------------------------------------------
        |C:96.7%[S:96.4%,D:0.3%],F:2.5%,M:0.8%,n:1013      |
        |980    Complete BUSCOs (C)                        |
        |977    Complete and single-copy BUSCOs (S)        |
        |3      Complete and duplicated BUSCOs (D)         |
        |25     Fragmented BUSCOs (F)                      |
        |8      Missing BUSCOs (M)                         |
        |1013   Total BUSCO groups searched                |
        ---------------------------------------------------
2023-09-19 15:40:10 INFO:       BUSCO analysis done. Total running time: 63 seconds
2023-09-19 15:40:10 INFO:       Results written in /workspace/busco/Exercise2/Pediculus
2023-09-19 15:40:10 INFO:       For assistance with interpreting the results, please consult the userguide: https://busco.ezlab.org/busco_userguide.html

2023-09-19 15:40:10 INFO:       Visit this page https://gitlab.com/ezlab/busco#how-to-cite-busco to see how to cite BUSCO
(BioDivBUSCO) gitpod /workspace/busco/Exercise2 (main) $
```

Question: Despite the "low" number of genes in this genome, what can we say about the completeness of this annotation set?

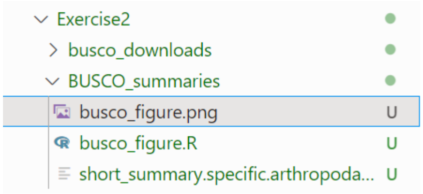
- If BUSCO failed, you can get the file from here instead, using 'wget':
- `wget https://apollo.vital-it.ch/trackvis/BGA23/Pediculus.zip`
- `unzip Pediculus.zip`

- Use the explorer to see the output files and folders, although they are similar to when we ran a genome assembly assessment, note this time there are no “metaeuk” results folders, only the sequences and the hmmer_output, along with the table, list and summary files

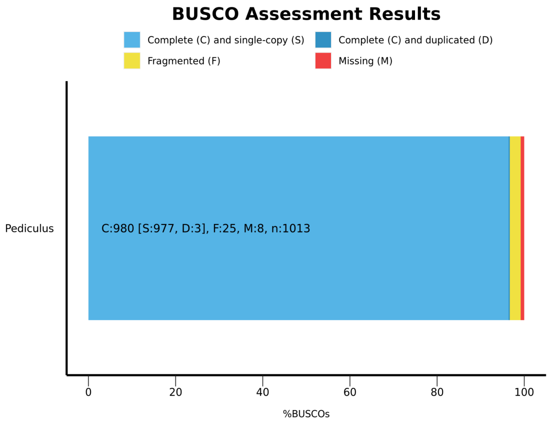


```
∨ Exercise2                                   ●
  > busco_downloads                           ●
  > ncbi_dataset                              ●
  ∨ Pediculus                                 ●
    > logs                                    ●
    ∨ run_arthropoda_odb10                    ●
      > busco_sequences
      > hmmer_output                          ●
        full_table.tsv                        U
        missing_busco_list.tsv                U
      {} short_summary.json                   U
        short_summary.txt                     U
    {} short_summary.specific.arthropoda...   U
       short_summary.specific.arthropoda...   U
```

- If you want, you can once again plot the results of the BUSCO run:
- `mkdir BUSCO_summaries`
- `cp Pediculus/short_summary.specific.arthropoda_odb10.Pediculus.txt BUSCO_summaries/`
- `python3 /workspace/conda/envs/BioDivBUSCO/bin/generate_plot.py -wd BUSCO_summaries`

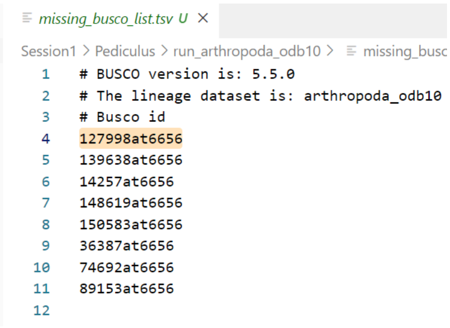


```
∨ Exercise2                                   ●
  > busco_downloads                           ●
  ∨ BUSCO_summaries                           ●
      busco_figure.png                        U
      busco_figure.R                          U
      short_summary.specific.arthropoda...    U
```



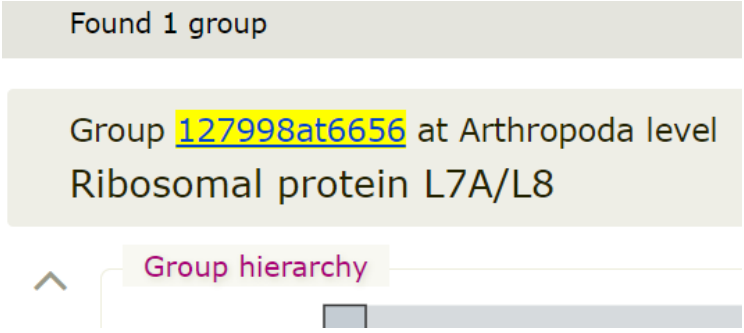
Question: How many “missing” BUSCOs are reported for this annotation set?

- Let’s explore OrthoDB to investigate what these missing genes might be, e.g. searching OrthoDB v10 for the first one, 127998at6656, shows it to be Ribosomal protein L7A/L8, an otherwise highly-conserved gene (168 of 172 arthropods)



missing_busco_list.tsv U ×
Session1 > Pediculus > run_arthropoda_odb10 > missing_busc

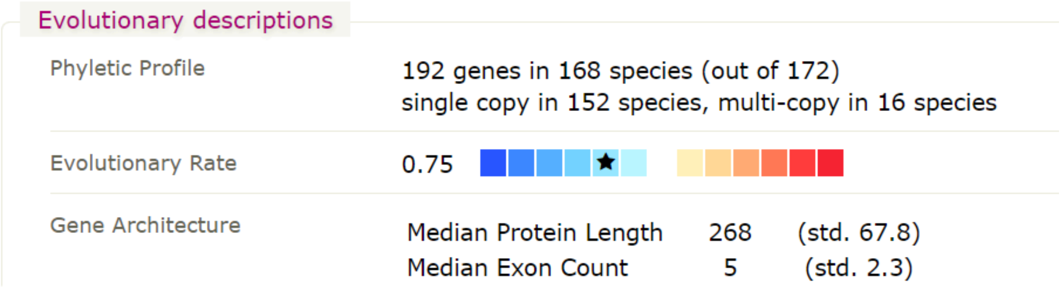
```
# BUSCO version is: 5.5.0
# The lineage dataset is: arthropoda_odb10
# Busco id
127998at6656
139638at6656
14257at6656
148619at6656
150583at6656
36387at6656
74692at6656
89153at6656
```



Found 1 group

Group 127998at6656 at Arthropoda level

Ribosomal protein L7A/L8

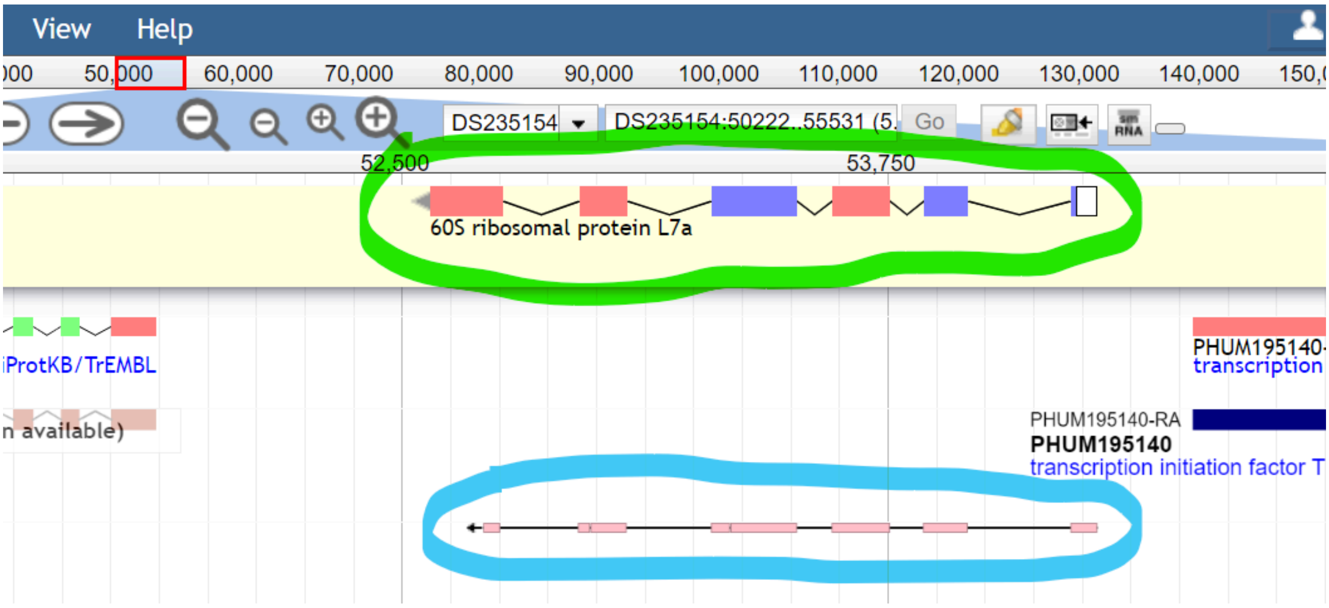


Evolutionary descriptions

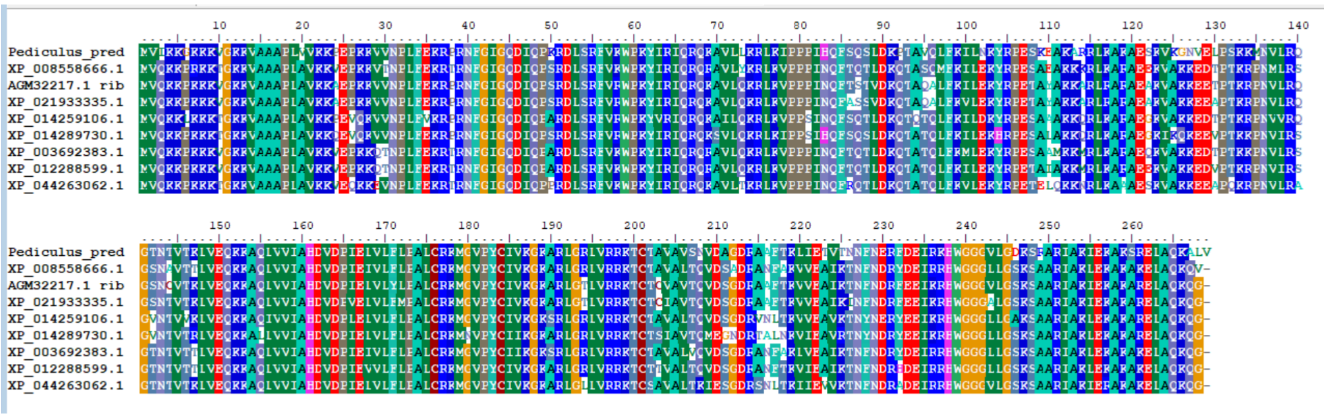
Phyletic Profile	192 genes in 168 species (out of 172) single copy in 152 species, multi-copy in 16 species
Evolutionary Rate	0.75
Gene Architecture	Median Protein Length 268 (std. 67.8) Median Exon Count 5 (std. 2.3)

- I thought that this was probably unlikely to be a real missing gene, so I searched the genome using the *Drosophila melanogaster* orthologue and I found a good hit in the *Pediculus* genome assembly, in a region where no gene was annotated!

- I therefore used the available transcriptomics data (blue) to create a manual annotation for this gene (green), so one day when manual annotations make their way into official gene sets at the NCBI this gene will no longer appear as “missing” when performing an assessment of the annotated gene set



- A sequence alignment with orthologues from other species convinces me that I have managed to correctly annotate the complete gene in *Pediculus*.



<u>Question:</u> What can we conclude from this exploration of the genome?

- A cautionary note about alternative transcripts
  - We used the *Pediculus* annotation from the NCBI
  - We did no pre-filtering of this annotation set to select just one protein representative per gene

<u>Question:</u> Why would we normally want to select just one protein representative per gene when running an assessment of an annotation set?

- We can actually count the number of proteins in the *Pediculus* protein FASTA file to see if this affects our analysis:
- `grep -c '>' ncbi_dataset/data/GCF_000006295.1/protein.faa`

- This tells us there are 10’775 proteins in this file, but remember that the NCBI page indicated that there were 10’773 protein-coding genes in this annotation set:
  - In this case therefore there is at least one gene, possibly two genes, with alternative transcripts annotated
  - So here the impact on our analysis will be negligible if anything

- The NCBI are working on providing “reference” annotation sets that contain one selected representative protein per gene. For now though you would have to perform the filtering yourself if you wanted to ensure the “duplication” values produced by BUSCO make sense in terms of reporting real gene duplications rather than alternative protein products of a single gene.

<u>Final Question:</u> If we were assessing instead a de novo transcriptome assembly (i.e. BUSCO in “transcriptome” mode), what do we need to think about when considering alternative transcripts of a single gene?

Reminders about GitPod

- Only files in the /workspace directory are kept between state transitions (when you stop and restart a workspace), so don’t store files anywhere else otherwise they will be deleted when you stop the workspace
- Always stop workspace before leaving, otherwise it will keep running and be billed!
    - ➢ 'Menu' (the 3 horizontal bars top-left of the window)
    - ➢ 'Gitpod: Stop workspace' (lower half of the menu)

Troubleshooting

- If you have a problem with your browser not showing everything, blocking pop-ups...: https://www.gitpod.io/docs/configure/user-settings/browser-settings
- If you have a problem when authenticating through gitlab, github or bitbucket: https://www.gitpod.io/docs/configure/authentication
- If the text you enter in the gitpod terminal is invisible, try deactivating the GPU acceleration:
    - ➢ Click on the cog in the bottom-left
    - ➢ ‘Settings’
    - ➢ Type ‘gpu acceleration’ in the search bar
    - ➢ Set ‘Terminal › Integrated: Gpu Acceleration’ to off’

Congratulations!