ColabCuraTE v1.0.1

An easy-to-use pipeline for manual curation of transposable elements

INTRODUCTION

This colab notebook is a pipeline for the manual curation of transposable elements (TEs).

Requirements:

- A genome assembly file (*fasta* format) from an organism of interest.
- A repeat library file (*fasta* format) from the same organism, generated using a TE discovery and annotation tool (e.g., the `XXX-families.fa` file output from *RepeatModeler2*).
- The name of a specific TE family from the repeat library to curate

Output:

- An improved, full length consensus sequence (*fasta* format) for the TE family of interest.
- An updated *stockholm* formatted (`.stk`) seed alignment file which can be deposited in the *Dfam* TE database.

Pipeline steps:

1. Load input files and install software packages.
2. Select a TE family of interest and extract, align, and extend individual copies
    - Extract a TE consensus sequence from the repeat library.
    - Identify all of its copies in the genome.
    - Align TE copies to create a *seed alignment*.
    - Perform initial extension of TE copy boundaries
3. Align and inspect extended copies
    - Create a new *seed alignment* from extended TE copies
    - Visualize the alignment to identify TE boundaries
    - Iteratively extend TE copies until boundaries are identified
4. Precise identification of TE boundaries
    - Search for Target Site Duplications (TSDs) to help identify exact TE boundaries

- Trim the alignment down to the TE boundaries
- Create a new, extended, consensus.

5. Generation of final output files
   - Reclassify the final consensus sequence.
   - Create a final *stockholm* formatted (`.stk`) seed alignment ready for *Dfam* accessioning.
6. Analysis of consensus sequence
   - Compare the original and updated consensus sequences
   - Analysis of updated consensus sequence with *TE-Aid*
7. Appendix
   - Running *RepeatModeler2* and *RepeatMasker* on *Galaxy*
   - Creation of repeat landscape plots
   - Generate interactive table of summary statistics for all TE families

PART 1. LOAD FILES & INSTALL SOFTWARE PACKAGES

1.1) COPY FILES FROM GOOGLE DRIVE

Run this cell to mount your google drive to this notebook. Follow the instructions in the popup windows to allow access.

> ***IMPORTANT***: To successfully mount your google drive, you must click ***Select all*** when prompted to ***Select what Google Drive for desktop can access***. If you have concerns about allowing this notebook to access your google drive, we recommend that you either create a new google account that you use only for *Colab* or follow the ***Direct Upload*** instructions below.

Mounting your **Google Drive** is beneficial for two key reasons:

1. **More reliable uploads**: Transfer of large genome files directly from your local device can be interrupted, resulting in a truncated file. Uploading from Google Drive avoids this issue.
2. **Automatic file saving**: Several steps save backup copies of important files to your Google Drive. These backups help prevent data loss if the Colab runtime

times out after the step completes.

Required Google Drive File Structure

Before starting, please create the following folders inside your main `MyDrive` folder:

```
MyDrive/TE_curation_files
MyDrive/TE_curation_files/genomes
MyDrive/TE_curation_files/repeat_libraries
MyDrive/TE_curation_files/stks
MyDrive/TE_curation_files/extended_alignments
MyDrive/TE_curation_files/curated_families
```

Place your **genome file** in: `MyDrive/TE_curation_files/genomes/`

Place your **repeat library file** in: `MyDrive/TE_curation_files/repeat_libraries/`

> *If you've previously run Step 2.2 (seed alignment), you can use the step below to load an autosaved `.stk` alignment file from: `MyDrive/TE_curation_files/stks`*

> *If you've previously run Step 3 (alignment extension), you can use the step below to load an autosaved `.fa` extended alignment file from: `MyDrive/TE_curation_files/extended_alignments`*

Your google drive file structure should now appear as below:

```
MyDrive/
└── TE_curation_files/
    ├── genomes/
    │   └── your_genome.fasta
    ├── repeat_libraries/
    │   └── your_repeat_library.fasta
    ├── stks/
    │   └── your seed alignments will be autosaved here
    ├── extended_alignments/
    │   └── your extended alignments will be saved here
    └── curated_families/
        └── your final curated TE family files will be saved here
```

Show code

```
Mounted at /content/drive
```

Run this cell to copy your files over from Google Drive

- Edit below to specify the file names of your genome and repeat library in your Google Drive folders.

genome: GCA_023653725.1_ASM2365372v1_genomic.fna

repeat_library: Bungarus_multicinctus-families.fa

If you want to load a previously saved seed alignment from `MyDrive/TE_curation_files/stks`, enter the filename below, otherwise enter `None`.

seed_alignment: None

If you want to load a previously saved extended alignment from `MyDrive/TE_curation_files/extended_alignments`, enter the filename below, otherwise enter `None`.

extended_alignment: None

Show code

ALTERNATIVE TO STEP 1.1) DIRECT FILE UPLOAD (NOT PREFERRED)

If you prefer not to use Google Drive: you must manually upload your input files and manually download any output files you want to keep.

Direct upload instructions (genome & TE library):

1. Click the **folder** icon (left sidebar) to open *Files*.
2. Click **Upload** and select your **genome FASTA** and your **TE library FASTA** from your computer. While uploading, check the progress bar on the bottom left corner of this notebook to see upload status.
3. After upload, confirm the files appear in `/content/` and enter the genome name in the **Genome Size Check** box below. Then run this cell to confirm your genome was fully uploaded.

Direct download instructions:

- Open the **Files** panel (click the **folder** icon on the far-left sidebar).

- Find your file under `/content/`.
- Click the **three dots** next to the filename → **Download**.

Genome Size Check: After uploading your genome FASTA manually, enter the filename below to calculate the size of the uploaded genome. To confirm that the upload was not truncated, verify that the reported size matches the expected size.

genome: " Insert text here "

Show code

Please be aware of the following:

- Your input files and any output files generated during this pipeline will be stored in the `/content` directory of this notebook. Expand the folder icon on the left side of the notebook to see the files/folders present in the `/content` directory for this current runtime.
- All files in the `/content` directory of this notebook will be deleted when the runtime expires or is terminated, which occurs if the notebook is idle for 90 minutes or when the maximum session length (12 hours) is reached. We introduce several checkpoints below to back up important files but any file can be saved at any time. To save a file, expand the folder icon on the left side of the notebook, click the three dots that appear when you hover over the filename and select *Download*.
- It may take some time for the `/contents` folder to automatically refresh and display newly created files. In this case, use the *Refresh* icon to manually refresh the directory.

1.2) INSTALL SOFTWARE DEPENDENCIES

This installation will take a few minutes.

Show code

```
Installing pdf2image...
pdf2image installed successfully.
Installing biopython...
biopython installed successfully.
Installing mamba...
mamba downloaded successfully.
mamba installed successfully.
mamba configured successfully.
Installing dependencies using mamba...
```

```
Installing TE-Aid...
```

> *NOTE: Colab notebooks run on a virtual machine in the cloud by connecting to a remote runtime. If the notebook sits idle for 90 minutes, it will disconnect from the runtime. To start using the notebook again, run the code cells above to (re)install the software*

PART 2. EXTRACT & EXTEND TE COPIES

TE *de novo* prediction tools, such as *RepeatModeler2*, output all predicted TE consensus sequences combined in a single file (e.g. `XXX-families.fa`). The code below allows a single consensus sequence to be extracted from the combined `XXX-families.fa` file.

> *For example, in our RepeatModeler2 run, the Copia TE family consensus sequence from Drosophila melanogaster was assigned the ID `rnd-3_family-103`*

2.1) EXTRACT TE CONSENSUS SEQUENCE FROM THE REPEAT LIBRARY

- Edit this to specify the ID of the TE consensus sequence to extract from the repeat library file (*e.g.* `rnd-3_family-103`).

seqID: rnd-1_family-245

- Specify the name of the input repeat library FASTA file that contains the consensus sequences (*e.g.* `dmel-families.fa`).

infile: Bungarus_multicinctus-families.fa

- Specify the output FASTA filename (*e.g.* `copia_con.fa`).

outfile: gypsy_con.fa

Show code

```
Consensus sequence written to gypsy_con.fa
```

2.2) CREATE A NEW SEED ALIGNMENT USING *REPEATMASKER*

This will output a new seed alignment file (`*.stk`).

> Please note the following:
>
> - *RepeatMasker* can take several hours to run on gigabase-sized genomes. If this step does not successfully complete in the notebook, an alternative strategy is to run *RepeatMasker* on *Galaxy* by following the instructions in **Step 7.6**.
> - The most efficient approach for large-scale curation efforts on the same genome is to run *RepeatMasker* on *Galaxy* using the full set of TE consensi generated by *RepeatModeler2* (or similar software, see Appendix **Steps 7.1–7.2**). Once complete, use **Step 7.5** to extract individual seed alignments for each TE family. This allows you to skip **Step 2.2** here and proceed directly to **Step 2.3**.

- Edit below to specify the name of the FASTA file (from Step 2.1 above) containing your TE consensus sequence (*e.g.* `copia_con.fa`).

`TE_consensus:` " gypsy_con.fa "

- Edit below to specify the name of the genome assembly fasta file (*e.g.* `dmel.fasta`).

`genome:` " GCA_023653725.1_ASM2365372v1_genomic.fna "

- Edit below to specify the species name (*e.g. Drosophila melanogaster*).

`taxon:` " Bungarus multicinctus "

> *NOTE: Here we use the `-qq` and `-nolow` RepeatMasker arguments to speed up the process so that it runs quickly. Adjust the options below to control the tradeoff between speed and sensitivity.*

`speed:` " qq "

- `d`: Default search speed
- `s`: Slow search; 0-5% more sensitive, 2-3 times slower than default
- `q`: Quick search; 5-10% less sensitive, 2-5 times faster than default
- `qq`: Rush job; about 10% less sensitive, 4->10 times faster than default

nolow: " True "

- Use of the `nolow` option skips masking of low complexity DNA and simple repeats, which substantially reduces the runtime with the caveat that these sequences, when left unmasked, may be incorrectly identified as TEs. To include pre-masking of simple/low complexity DNA, set `nolow` to `False` above.

> *NOTE: RepeatMasker runs that exceed 12 hours will be terminated due to the limit on Colab session lengths. In this case, we recommend running RepeatMasker on Galaxy, as described in the Appendix.*

Show code

```
RepeatMasker run completed successfully.
Seed alignment written in Stockholm format to: Bungarus_multicinctus_rnd-1_fa
Backup copy saved in Google Drive: /MyDrive/TE_curation_files/stks/Bungarus_m
```

Checkpoint: Seed Alignment Output (`.stk` file)

Step 2.2 above can take **several hours** for large genomes. If you leave your computer to run (overnight, for example) the Colab session may time out due to inactivity after *RepeatMasker* finishes. If you linked your Google Drive, **Step 2.2** will **automatically save** your seed alignment `.stk` file to the `MyDrive/TE_curation_files/stks/` folder: thus, **you won't lose any data**. However, if you did not link your Google Drive, you must download the seed alignment manually before the session times out and the file is deleted.

If your session times out but your seed alignment was saved, follow the steps below:

- Click the *Reconnect* button on the top left to connect the notebook to a new runtime
- Rerun the **setup and installation cells** in Step 1.
- Enter the filename in Step 1.1 to copy your saved seed alignment `.stk` file from your Google Drive to your notebook
- Resume the pipeline by skipping to **Step 2.3** below

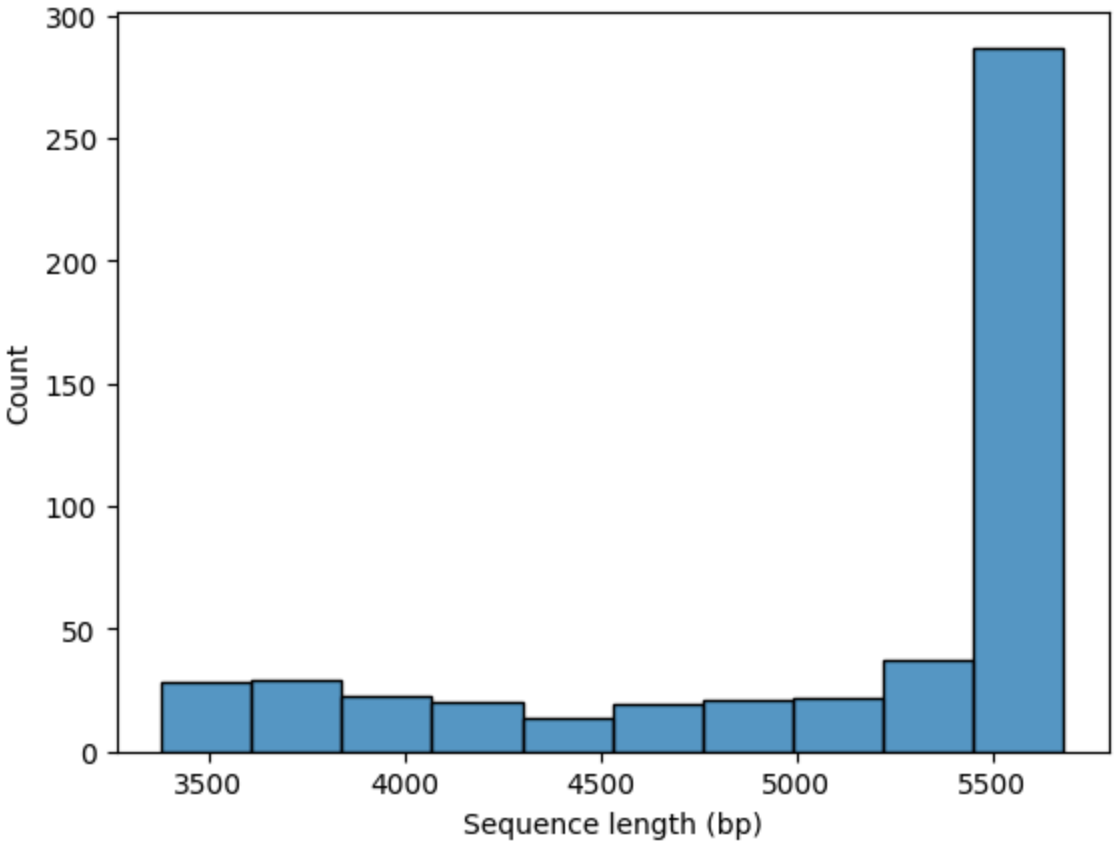
2.3) SELECT SEED LENGTH CUTOFF

> *Full length seed copies will be the most useful for generating the full length consensus sequence. Thus, one may want to exclude smaller seeds from the `.stk` file.*

The command below creates a histogram of seed lengths from the `.stk` file, which can be used to help select a length cutoff

- Specify the name of the `.stk` file that contains the seed alignment generated in Step 2.2 (*e.g.* `Drosophila_melanogaster_rnd-3_family-103.stk`).

`infile:` "Bungarus_multicinctus_rnd-1_family-245.stk"



2.4) FILTER SEED ALIGNMENT BY LENGTH

- Exclude sequences shorter than this length (*e.g.* `1500`). Set to `0` to select the `X` longest sequences where `X` = `max_sequences` below

`length_cutoff:` 5500

- Maximum number of sequences to keep (*e.g.* `50`).

`max_sequences:` 50

- Name of the *Stockholm* file that contains the seed alignments (*e.g.* `Drosophila_melanogaster_rnd-3_family-103.stk`).

infile: "Bungarus_multicinctus_rnd-1_family-245.stk"

- Output filename for new seed alignment (*e.g.* `Drosophila_melanogaster_rnd-3_family-103.filtered.stk`).

outfile: "Bungarus_multicinctus_rnd-1_family-245.filtered.stk"

```
Retained 50 sequences in Bungarus_multicinctus_rnd-1_family-245.filtered.stk
```

2.5) EXTEND TE SEED COPIES

Add flanking genome sequence to both ends of each TE copy in the length-filtered seed alignment. Extending TE copies ensures that the final consensus will represent the full length TE

- Specify the amount of nucleotide flanking sequence to add to each end (*e.g.* left and right flanks) of each seed. As a starting point, we suggest adding 200 nucleotides to each flank.

left_flank: 400

right_flank: 800

- Name of genome assembly fasta file (*e.g.* `dmel.fasta`).

genome: "GCA_023653725.1_ASM2365372v1_genomic.fna"

- Name of the filtered seed alignment file from *Step 2.4* (*e.g.* `Drosophila_melanogaster_rnd-3_family-103.filtered.stk`).

infile: "Bungarus_multicinctus_rnd-1_family-245.filtered.stk"

- Output seed filename in FASTA format with flanking sequence added (*e.g.* `copia_elements.fa`).

outfile: "gypsy_elements.fa"

Show code

```
Extended left flank by 400 bp and right flank by 800 bp
Output saved in gypsy_elements.fa
```

PART 3. ALIGN AND EXTEND TE COPIES

The TE seeds need to be extended until they include the entire length of the TE. The TE edges (i.e. the beginning and end of the TE) can be inferred by aligning the TE seeds and then plotting the conservation of each alignment column.

Starting from the left side of the plot, the point where the conservation score jumps from ~25% to above ~50% (closer to 100% for active TE families) marks the beginning of the TE. The conservation score should remain relatively high for the full length of the TE, although short decreases are possible. The end of the TE is denoted by the point where the conservation score drops back down to ~25%.

> *Note: if the conservation score is high across the entire alignment, this means that the seeds have not been extended far enough and additional extension is needed to reach the beginning and/or end of the TE (see Step 3.3).*

3.1) GENERATE SEED ALIGNMENT FROM EXTENDED COPIES

- Filename for the extended seed sequences from *Step 2.5* above (*e.g.* `copia_elements.fa`).

infile: gypsy_elements.fa

- Name of the alignment output file (*e.g.* `copia_elements_aligned.fa`).

outfile: gypsy_elements_aligned.fa

> *Note: This can take 30 minutes or more depending on the amount of flanking sequence added to the seeds.*

Show code

```
generating a scoring matrix for nucleotide (dist=200) ... done
Gap Penalty = -1.53, +0.00, +0.00
tbutree = 1, compacttree = 0
Constructing a UPGMA tree ...
   40 / 50
done.

Progressive alignment ...
STEP     2 /49
Reallocating..done. *alloclen = 14803
STEP    39 /49
Reallocating..done. *alloclen = 15854
STEP    49 /49
done.
tbfast (nuc) Version 7.526
alg=A, model=DNA200 (2), 1.53 (4.59), -0.00 (-0.00), noshift, amax=0.0
1 thread(s)

minimumweight = 0.000010
autosubalignment = 0.000000
nthread = 0
randomseed = 0
blosum 62 / kimura 200
poffset = 0
niter = 16
sueff_global = 0.100000
nadd = 16
Loading 'hat3' ... done.
generating a scoring matrix for nucleotide (dist=200) ... done

   40 / 50
Segment   1/  1    1-8300
done
dvtditr (nuc) Version 7.526
alg=A, model=DNA200 (2), 1.53 (4.59), -0.00 (-0.00), noshift, amax=0.0
0 thread(s)


Strategy:
 L-INS-i (Probably most accurate, very slow)
 Iterative refinement method (<16) with LOCAL pairwise alignment information

If unsure which option to use, try 'mafft --auto input > output'.
For more information, see 'mafft --help', 'mafft --man' and the mafft page.

The default gap scoring scheme has been changed in version 7.110 (2013 Oct).
It tends to insert more gaps into gap-rich regions than previous versions.
To disable this change, add the --leavegappyregion option.
```

3.2) VISUALIZE ALIGNMENT CONSERVATION

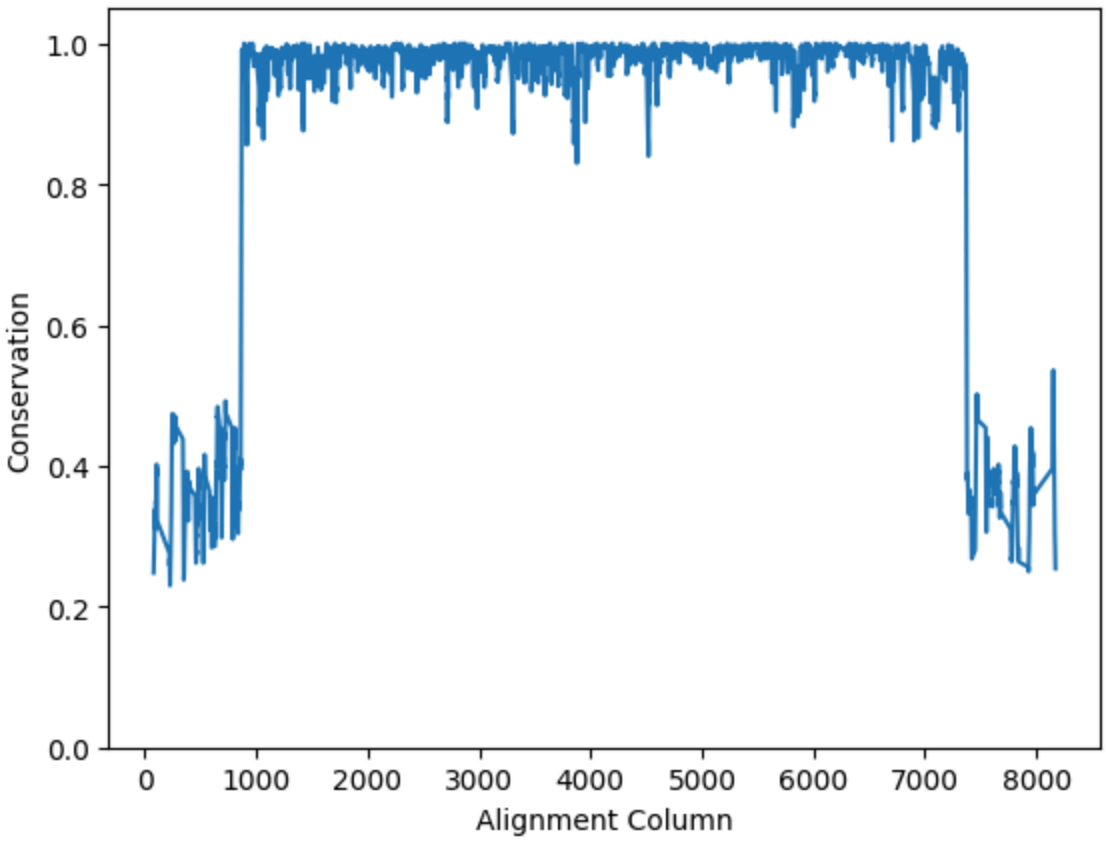
This cell plots the conservation score of each column of the seed alignment. A conservation score close to 1.0 reflects high sequence similarity at that position, indicating that those columns are part of the TE. A low conservation score (~0.25 or lower) corresponds to genomic regions that are no longer homologous (*i.e.* not part of the TE).

When the seed sequences have been extended enough to include the full length TE, the plot will show a contiguous region of highly conserved sequence flanked by regions of low sequence conservation on both ends.

If the plot shows highly conserved sequence on one or both edges, additional extension is needed. In this case follow *Step 3.3* below.

- Name of the alignment file from *Step 3.1* (*e.g.* `copia_elements_aligned.fa`).

infile: gypsy_elements_aligned.fa



3.3) OPTIONAL: ADDITIONAL EXTENSION OF SEED SEQUENCES

Follow the steps below if the conservation plot suggests that the TE seeds need to be extended further.

1. Return to *Step 2.5* and increase one or both flanks, depending on what the conservation plot shows. We suggest doubling the increment of flanking sequence you are adding each time (*i.e.* 200, 400, 800, 1600, ...) If the flank of one end of the TE becomes apparent in the plot, it is only necessary to extend the other flank.
2. Rerun *Steps 3.1 & 3.2*
3. Repeat until the conservation plot shows flanks of low conservation on both ends

Note: It is not necessary to modify the filenames for each iteration of extension.

Checkpoint: Back Up Your Alignment File After Extension Is Complete

You've now completed the **iterative extension and alignment process**. Before continuing to **Part 4**, we recommend that you **back up** your final extended alignment file (`.fa`) to `MyDrive/TE_curation_files/extended_alignments` by **running the cell below if you linked your Google Drive**. This will prevent you from having to repeat these steps if your session times out due to inactivity or if you plan to pause and resume the curation workflow later.

If you **did not link Google Drive, you'll need to download the file manually**.

To resume later:

1. Rerun the **setup and installation** steps in a new Colab session
2. **Copy** your saved extended alignment file from Google Drive in Step 1.1
3. Continue with **Part 4** of the pipeline

`final_alignment:` gypsy_elements_aligned.fa

Show code

```
Saved to Google Drive: MyDrive/TE_curation_files/extended_alignments/gypsy_el
```

PART 4. PRECISE IDENTIFICATION OF TE BOUNDARIES

4.1) VISUALIZE SEED ALIGNMENT TO IDENTIFY THE EDGES OF THE TE

This cell generates an interactive alignment visualization tool. After running the cell, use t
settings dropdown menu to adjust the following parameters:

```
color scheme: clustalx_dna   ← you can switch to a different scheme if you
row height: 8                ← increase/decrease to zoom in/out vertically
column width: 8              ← increase/decrease to zoom in/out horizontall
```

> *Note: hovering your mouse over the column will interactively identify its position in the alignment, which is necessary for specifying the TE edges in the following steps.*

- Specify the name of the seed alignment file from *Step 3.1* (*e.g.* `copia_elements_aligned.fa`).

`alignment:` " gypsy_elements_aligned.fa "

Show code

To find the TE edges, scroll through the alignment to the point where the sequences become clearly alignable. This point represents the beginning of the TE. If you have difficulty finding this region, we recommend zooming out by reducing the `row height` and `column height` parameters in the settings dropdown menu, as described above.

> *Note: the TE edges may or may not be clearly defined, depending on the properties of the TE family being investigated. The goal here is to identify a range of alignment columns that contain the TE boundary, for both the left and right edges of the TE. Use a lenient range to be sure that the true TE edge is included within each span. The Target Site Duplication step below will help refine the edges. If the alignment does not show ambiguous/unalignable sequence on both ends, further extension of seed sequences is needed. Return to Step 3.3 in this case.*

Record the column numbers that span the beginning of the TE, where the sequences start to align. The first column in the range should lie within the unaligned sequence >= 20 nucleotides before the beginning of the TE. The second column should lie 5-10 nucleotides after the beginning of the TE. This range will be used for the `left_start` and `left_end` parameters in the trimming step below.

Continue scrolling through the aligned sequences until reaching the end of the TE, where the sequences are no longer clearly aligned to one another. Record the column numbers that span the end of the TE. The first column in the range should lie within the aligned sequence, 5-10 nucleotides before the point where the sequences are no longer aligned. The second column should lie >=20 nucleotides after the end of the TE. This range will be used for the `right_start` and `right_end` parameters in the trimming step below.

4.2) SEARCH FOR TARGET SITE DUPLICATIONS (TSDs)

Running this cell removes the internal portion of the seed alignment to allow visualization (in *Step 4.3*, below) of just the left and right TE edges, separated by `XXX`

- Specify the range of columns for the left edge of the TE, as explained in *Step 4.1* above.

left_start: 842

left_end: 867

- Specify the range of columns for the right edge of the TE, as explained in *Step 4.1* above.

right_start: 7364

right_end: 7389

- Specify the filename of the seed alignment from *Step 3.1* (*e.g.* `copia_elements_aligned.fa`).

alignFile: "gypsy_elements_aligned.fa"

- Specify an output filename for the internally-trimmed alignment that includes only the ranges specified above (*e.g.* `alignment_ends.fa`).

outfile: "alignment_ends.fa"

```
Internally trimmed alignment written to: alignment_ends.fa
```

4.3) SEARCH FOR TARGET SITE DUPLICATIONS (TSDs) & IDENTIFY EXACT TE EDGES

This cell visualizes the internally trimmed alignment from *Step 4.2*, which makes it easier to identify TSDs

Some TEs are flanked by a constant-length TSD as a result of their integration mechanism. Examine the alignment to search for repeated sequences that appear just before, and just after, the TE edges. In the example below, a 5bp TSD is shown for a TE represented by the string of `T's`.

> *Note: Target site duplications directly flank the beginning and end of a TE but are NOT part of the TE.*

```
...AGCGATTTTTTTTTTTTTTTTTTTAGCGA...
...CGTTATTTTTTTTTTTTTTTTTTTCGTTA...
...ATCAGTTTTTTTTTTTTTTTTTTTATCAG...
...GGCATTTTTTTTTTTTTTTTTTTTGGCAT...
```

> *Most TSDs range from 2 to 20 basepairs in size. The sequence that is duplicated will be unique for each TE copy (i.e. alignment row) but the*

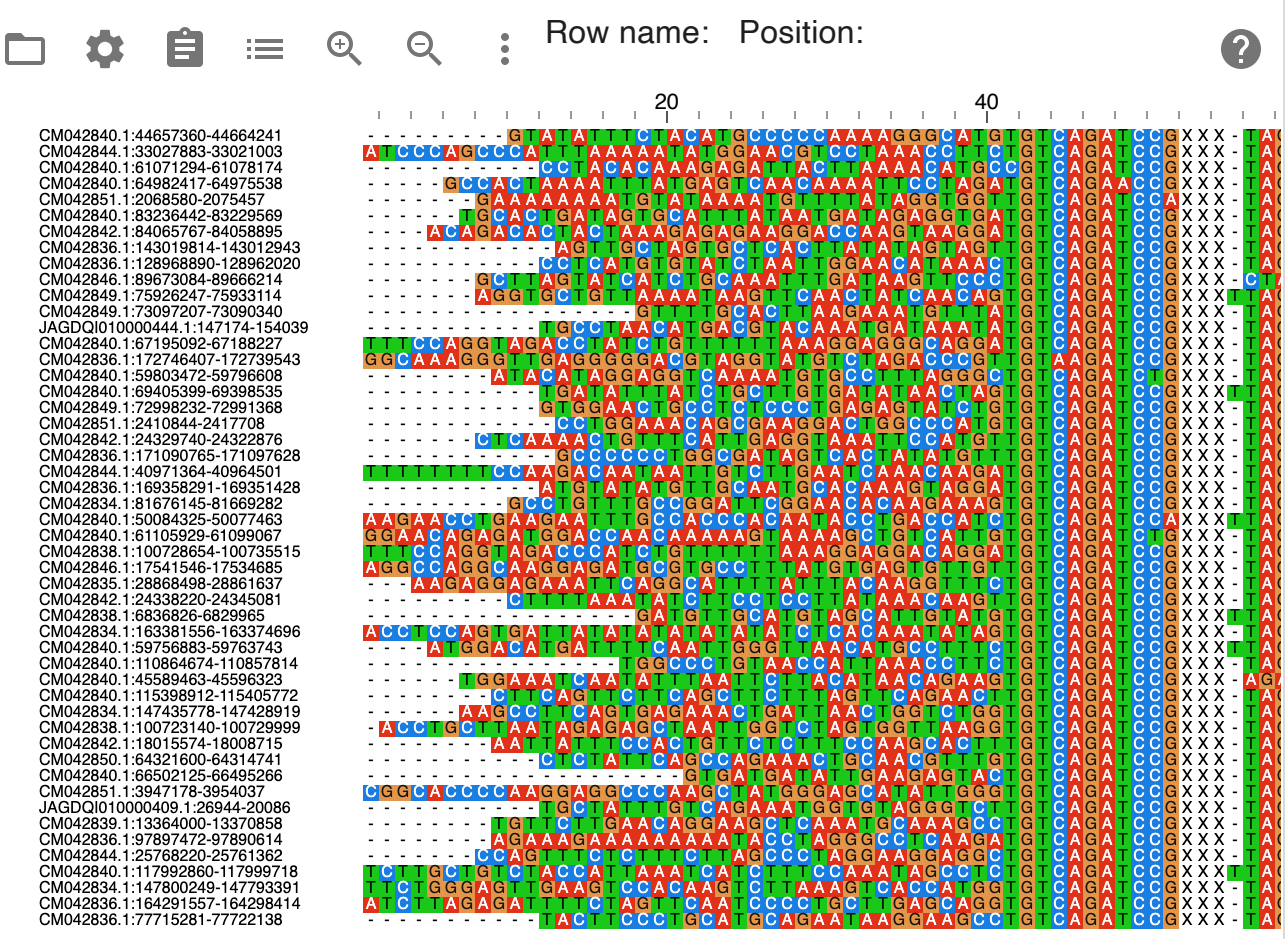
> *duplication length should be consistant across rows. Note that some TE families do not create TSDs at all.*

If you find a TSD, use its location to help define the exact columns where the TE starts and ends. If you cannot find a TSD, base your estimate on the first and last columns of the TE that show unambiguous alignment and good conservation.

- Name of the internally-trimmed alignment file from *Step 4.2* (*e.g.* `alignment_ends.fa`).

`alignment:` "alignment_ends.fa"

Show code



4.4) FINAL SEED ALIGNMENT TRIM

Provide the numbers for the exact start and end columns that encompass the complete TE in the seed alignment.

> *NOTE: Column numbers should be from the internally-trimmed (i.e. XXX-containing) alignment visualized in Step 4.3.*

- Enter the column where the TE starts

beg: 41

- Enter the column where the TE ends

end: 65

- Filename for the final trimmed alignment

outfile: "final_trimmed.fasta"

Show code

```
861 7369
```

4.5) CLEAN SEED ALIGNMENT TO REMOVE GAPPY REGIONS

Individual TE copies can acquire sequence insertions that result in gaps being added to all other sequences in the alignment. This step removes inserted sequences that are present in less than a specified fraction of the TE copies in the seed alignment.

- Filename of the trimmed alignment from *Step 4.4* above (*e.g.* `final_trimmed.fasta`).

infile: "final_trimmed.fasta"

- Output filename for the cleaned alignment (*e.g.* `final_trimmed_cleaned.fasta`).

outfile: "final_trimmed_cleaned.fasta"

- Desired gap threshold (recommended default is **0.5**)

gap_threshold: 0.5

- **Gap threshold** controls which gap-containing columns are removed from the alignment.
  - A **value of 0.5** (recommended) removes columns with gaps present in **≥50%** of TE copies.
  - Setting it **higher (e.g. 0.7)** will remove **more** gap-containing columns (*i.e.* all columns with gaps present in ≥30% of sequences).

- Setting it **lower (e.g. 0.3)** will remove **fewer** gap-containing columns (*i.e.* only columns with gaps present in ≥70% of sequences).

Show code

```
Cleaned alignment written to final_trimmed_cleaned.fasta
```

PART 5. GENERATE FINAL FILES

5.1) FINAL CONSENSUS SEQUENCE

This step creates a single consensus sequence in FASTA format from the cleaned seed alignment.

- Filename of the cleaned alignment from *Step 4.5* (*e.g.* `final_trimmed_cleaned.fasta`).

infile: final_trimmed_cleaned.fasta

- Output filename for the consensus sequence (*e.g.* `copia_consensus.fasta`).

outfile: gypsy_consensus.fasta

Show code

```
  0 s       1 Mb (  0%) Reading alignment
  0 s       2 Mb (  0%) Reading alignment done (0 s).
50 seqs, length 6090
```

5.2) RECLASSIFY FINAL CONSENSUS SEQUENCE

Improving/refining the TE consensus sequence may improve our ability to classify it. This step runs the program `RepeatClassifier` to check if the classification of the consensus sequence changes. The output will be provided as a `.classified` file (*e.g.* `copia_consensus.fasta.classified`).

This output file is in FASTA format and contains the same DNA sequence as in the input file, however the ID is modified to reflect the classification and the sequence is reverse complemented if it matches known TE sequences on the minus strand.

- Input filename for the updated consensus sequence file from *Step 5.1* (*e.g.* `copia_consensus.fasta`).

consensus_file: " gypsy_consensus.fasta "

```
RepeatClassifier Version 2.0.5
=====================================
  - Looking for Simple and Low Complexity sequences..
  - Looking for similarity to known repeat proteins..
  - Looking for similarity to known repeat consensi..
=====================================
Classification complete. Results:
>final_consensus#LTR/Gypsy


IMPORTANT: RepeatClassifier detected that your consensus sequence is in the c
```

5.3) FINAL FILES: SEED ALIGNMENT (STK) AND CONSENSUS (FASTA)

This cell will save your trimmed seed alignment, along with metadata describing your TE, in the Stockholm format. The consensus sequence from *Step 5.2* will also be renamed to reflect the family name specified below. These two files can be deposited in the DFAM database.

- TE family name in *RepBase* format: `Superfamily-Number_genusSpecies` (*e.g.* `Copia-1_dMel`).

family_name: " Gypsy-1_bMul "

- Short description of the TE (*e.g.* `LTR retrotransposon with 5 bp TSD`).

description: " LTR retrotransposon with 5 bp TSD "

- Replace `Doe J` below with your Lastname and First Initial(s)

authors: " Doe J "

- TE classification in DFAM format (*e.g.* `Interspersed_Repeat;Transposable_Element;Class_I_Retrotranspos` `Copia`).

classification: " Interspersed_Repeat;Transposable_Element;Class_I_Retrc "

- Genus of TE host (*e.g. Drosophila*).

genus: "Bungarus"

- Species of TE host (*e.g. melanogaster*).

species: "multicinctus"

- Genome assembly accession/version (*e.g.* `dm6`).

assembly: "GCA_023653725.1"

- Final trimmed alignment output from *Step 4.5* (*e.g.* `final_trimmed_cleaned.fasta`).

trimmed_alignment: "final_trimmed_cleaned.fasta"

- Classified consensus from *Step 5.2* (*e.g.* `copia_consensus.fasta.classified`).

consensus: "gypsy_consensus.fasta.classified"

```
Final annotated seed alignment written to: Gypsy-1_bMul.stk
Final renamed consensus sequence written to: Gypsy-1_bMul.fasta
Saved both files to Google Drive: MyDrive/TE_curation_files/curated_families/
```

PART 6. TE SEQUENCE ANALYSIS

6.1) COMPARE THE ORIGINAL AND UPDATED CONSENSUS SEQUENCES

A dotplot shows where two sequences align. The amount by which the updated consensus was extended can be inferred based on where the original and updated consensus sequences align to each other.

- Filename of the original consensus sequence from *Step 2.1* (*e.g.* `copia_con.fa`).

original_consensus: "gypsy_con.fa"

- Filename of the final consensus sequence from *Step 5.3* (*e.g.* `Copia-1_dMel.fasta`).

`updated_consensus:` " Gypsy-1_bMul.fasta "

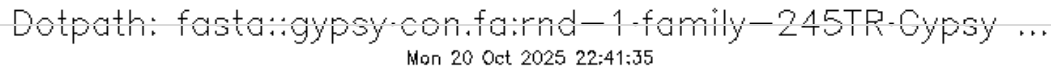
- Sequence similarity window size (as a default we recommend setting this to `10`).

`window_size:` 10

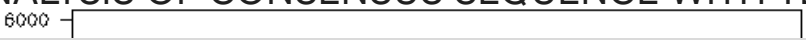
- Enter `True` below if *RepeatClassifier* reverse-complemented the consensus sequence in Step 5.2 above. Otherwise leave as `False`. If set to `True`, the dotplot will display the updated consensus sequence in its original orientation, prior to running *RepeatClassifier*.

`reverse_complemented:` " False "

```
Draw a non-overlapping wordmatch dotplot of two sequences
Created dotpath.1.png
```



6.2) ANALYSIS OF CONSENSUS SEQUENCE WITH TE-AID



TE-AID produces four plots:

1. (top left) Each horizontal line represents a single TE copy/fragment. The plot shows which part of the consensus sequence is covered by each copy and the sequence identity between them.
2. (top right) Genomic coverage of the consensus: TEs whose copies are frequently truncated will show differences in coverage across the consensus
3. (bottom left) Dotplot comparing the consensus to itself. This plot is useful for identifying structural features such as Long Terminal Repeats (LTRs) or Terminal Inverted Repeats (TIRs)
4. (bottom right) Structural and coding features including: TIR and LTR suggestions, open reading frames (ORFs) and TE protein hit annotations.

- Filename of the final consensus sequence from *Step 5.3* (*e.g.* `Copia-1_dMel.fasta`).

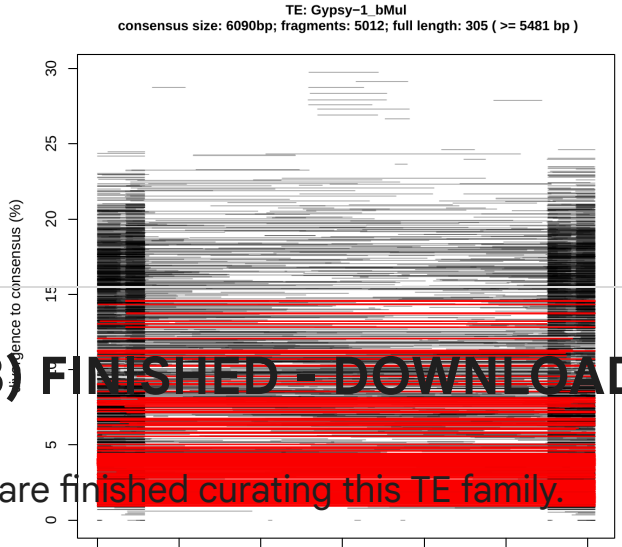
`final_consensus:` "Gypsy-1_bMul.fasta"

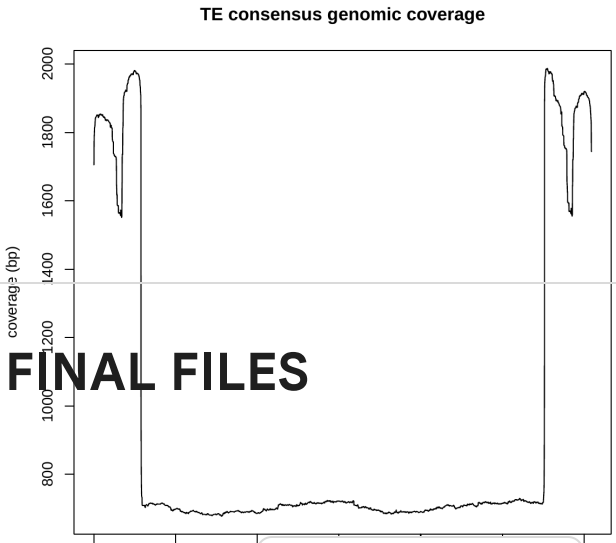
- Filename for the genome assembly (*e.g.* `dmel.fasta`).

`genome:` "GCA_023653725.1_ASM2365372v1_genomic.fna"

Show code

```
query:                          Gypsy-1_bMul.fasta
ref genome:                     GCA_023653725.1_ASM2365372v1_genomic.fna
TE -> genome blastn e-value:    10e-8
full length min ratio:          0.9
hits transparency:              0.3
full length hits transparency: 0.9
RepeatPeps.lib is not found, downloading...
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Curre
                                 Dload  Upload   Total   Spent    Left  Speed
100 17.1M  100 17.1M    0     0  45.8M      0 --:--:-- --:--:-- --:--:-- 45.9
Formating database
Blastp-ing...
WARNING: ignoring environment value of R_HOME
[1] "R: ploting genome blastn results and computing coverage..."
[1] "consensus length: 6090 bp"
[1] "R: ploting self dot-plot and orf/protein hits..."
null device
          1
Done! The graph (.pdf) can be found in the output folder: .
```

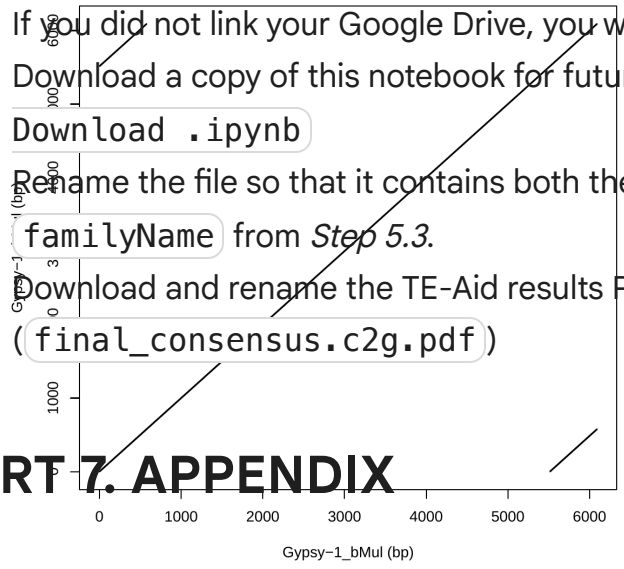


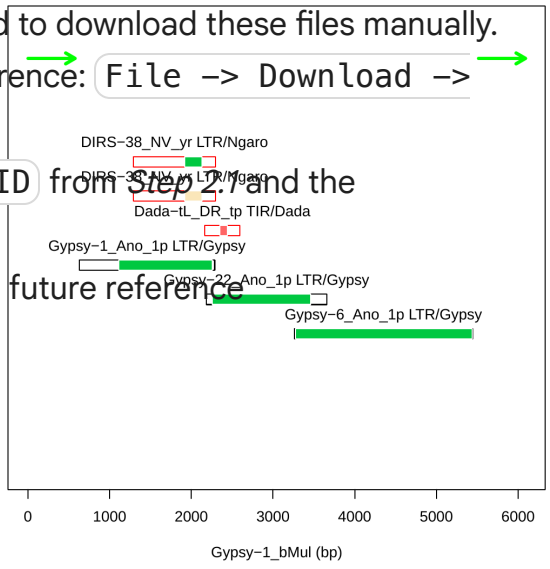


6.3) FINISHED - DOWNLOAD FINAL FILES

You are finished curating this TE family.

- If you linked your Google Drive, make sure your final files (*e.g.* `Copia-1_dMel.stk` and `Copia-1_dMel.fasta`) are present in your Google Drive in the folder `MyDrive/TE_curation_files/curated_families`
- If you did not link your Google Drive, you will need to download these files manually.
- Download a copy of this notebook for future reference: `File -> Download -> Download .ipynb`
- Rename the file so that it contains both the `seqID` from *Step 2.7* and the `familyName` from *Step 5.3*.
- Download and rename the TE-Aid results PDF for future reference (`final_consensus.c2g.pdf`)





PART 7. APPENDIX

Below are instructions for:

1. Using the Galaxy server to run *RepeatModeler2*
2. Using the Galaxy server to run *RepeatMasker*
3. Creation of a repeat landscape plot
4. Creation of an interactive repeat library summary table
5. Alternative *RepeatMasker* approach for large-scale curation efforts
6. Alternative *RepeatMasker* approach for extremely large genomes

> *If you do not already have a repeat library for your genome assembly and/or a TE family of interest for manual curation, you can follow Steps 7.1 - 7.4 to generate a de novo repeat library for your genome assembly and select a TE family of interest for manual curation.*

> *If you are curating many TE families from the same genome, it is more time-efficient to use the approach detailed in Step 7.5*

> *If the RepeatMasker step exceeds the allowable runtime in the notebook, you can use Galaxy instead, which allows a longer runtime, as described in Step 7.6*

7.1) Running **RepeatModeler** on the Galaxy Server

ColabCuraTE relies on a pre-existing **repeat library** as input, typically generated by *de novo* TE discovery tools such as **RepeatModeler2**. Running such tools on a whole-genome assembly is extremely computationally intensive and thus not feasible within the Google Colab environment. Below we provide details on how to use the **Galaxy web server** as a complementary resource to run *RepeatModeler2* for generating a repeat library.

We recommend running ***RepeatModeler2*** on the Galaxy Europe server, which allows for longer runtimes than other Galaxy servers.

Steps to Run *RepeatModeler2* on Galaxy:

1. **Go to:** https://usegalaxy.eu
2. **Create an account** or log in (required to submit jobs and track progress).
3. Upload your genome FASTA file:
    - Click the upload icon (top left).
4. In the Galaxy Tools panel, search for **"RepeatModeler"** and select it.
5. Set the parameters:
    - **Input Genome**: Select your uploaded FASTA file.
    - Leave other settings as default unless you have specific options.

6. Click **Run Tool** and wait for the job to complete (runtime may take several days depending on genome size). You'll be able to monitor progress in the history panel (right side).
7. Once complete, download the **RepeatModeler repeat library** file (usually named `RepeatModeler on data 1: consensus sequences`) for use as an input file in the *ColabCuraTE* pipeline.

7.2) Running **RepeatMasker** on the Galaxy Server

Running RepeatMasker on large genomes or with large repeat libraries (e.g. a full RepeatModeler output) can exceed the runtime limits of this notebook. Instead, we recommend running RepeatMasker on the Galaxy Europe server: https://usegalaxy.eu, which supports longer jobs and large datasets.

Follow These Steps:

1. **Sign in** to your account at https://usegalaxy.eu. If you don't have one, you'll need to create one (it's free).
2. **Upload your input files**:
   - A FASTA file of your organism's genome.
   - A FASTA file of your repeat library (e.g., the file output by RepeatModeler in Step 7.1).
   - Use the **Upload Data** button (top left) and select the files from your computer. Upload time will vary based on file size.
3. After the files finish uploading, search for **RepeatMasker** in the **Tools** panel (top left), and click the tool when it appears.
4. Set the parameters as follows:
   - **Genomic DNA**: Select your uploaded genome FASTA file.
   - **Repeat library source**: Choose `Custom library of repeats`.
   - **Custom library of repeats**: Select the repeat library file you uploaded.
   - **Advanced options → Output alignments file**: Set this to `Yes` to generate the seed alignment file.
   - Leave other settings as default unless you have specific options.
   - (Optional) Under **Email notification**, you can choose to be notified when the job completes.
5. Click **Run Tool** to start the job. Note: This may take several hours or longer depending on your genome and repeat library size.

6. When the job finishes, your output files will appear in the **History** panel (right side). The file you need is labeled:#@markdown

> `RepeatMasker alignment on data X and data Y`

7. Follow the instructions below to upload this file to the current Google Colab environment.

7.2 cont'd) Upload **RepeatMasker** alignment file from the Galaxy Server

If you processed data on Galaxy Europe, you can download your output files directly into this Colab notebook:

1. In Galaxy, go to the **History panel** on the right.
2. Click the **History options icon** (three horizontal lines) → click **"Share & Manage Access"**.
3. Click **"Make History accessible"** and save.
4. To upload the RepeatMasker alignment file, click its filename in the Galaxy server to expand it (*i.e.* `Repeatmasker alignment on data X and data Y`), and then click the **"Share link"** button that looks like a paperclip. This will copy the file link.
5. Paste this link in the space below and run this cell to upload the file from Galaxy.

- Specify the name of the link from Galaxy to the RepeatMasker alignment file (*e.g.* `https://usegalaxy.eu/api/datasets/26c75dcccb616ac84ceecc507717to_ext=txt`).

`galaxy_link:` " https://usegalaxy.eu/api/datasets/26c75dcccb616ac84ceecc5 "

The RepeatMasker alignment file will be saved as `repeatmasker.align` in the `/content` directory here.

Show code

7.3) Run **RepeatMasker** scripts to generate input files for assessing repeat landscape of your genome

This will generate a table summarizing the RepeatMasker output, from which you can select a TE family of interest to proceed with for curation. For instance, you may be interested in the most abundant TE families in this organism's genome. To do this, you

would sort the table by `Total Basepairs in Genome`, which will rank TE families by their abundance.

NOTE: This step may take ~30 minutes to run.

- Edit this to specify the name of the genome fasta file (*e.g.* `dmel.fasta`).

genome: "dmel.fasta"

- Edit this to specify a name for the output `*.divsum` file that this step will create (*e.g.* `dMel.divsum`).

divsum_file: "dMel.divsum"

- Edit this to specify a name for the output `*.html` file that this step will create (*e.g.* `dMel.html`).

html_file: "dMel.html"

Show code

7.3 cont'd) Create repeat landscape plot

This will generate a graph summarizing the repeat landscape from your organism's genome.

- Edit this to specify the filename for the `*.html` file that was output above (*e.g.* `dMel.html`).

html_file: "dMel.html"

Show code

7.4) Generate table summarizing **RepeatMasker** output

This will generate a table summarizing the RepeatMasker output, from which you can select a TE family of interest for manual curation. For instance, you may be interested in the most abundant TE families in your organism's genome. To do this, you would sort the table by `Total Basepairs in Genome`, which will rank TE families by their abundance.

- Specify the name of the `*.divsum` file output from the step above (*e.g.* `dMel.divsum`).

`divsum_file:` " dMel.divsum "

- The `index` column is the name of the TE (needed for `Step 2.1` in the curation notebook).
- Each of the other columns can be sorted by clicking on the header.
- The table rows can be filtered by clicking on the filter button.

Show code

7.4 cont'd) **Proceed to TE curation**

- Once you have selected your TE family of interest, you can proceed to **Step 2.1** in this pipeline (the seqID used in Step 2.1 is in the column labeled `Index` in the summary table above).
- If you have already run the earlier steps in this Appendix and have a `repeatmasker.align` file in your `/content` folder, you can skip **Step 2.2**. Instead, run the cell below (**Step 7.5**) to generate a seed alignment file (`.stk`) for your TE family of interest. This `.stk` file can then be used as input for **Step 2.3**.

Show code

7.5) Extract seed alignment from ***RepeatMasker*** full TE library output (**fastest strategy for large-scale curation efforts**)

The most efficient approach for large-scale curation projects is to run *RepeatMasker* on Galaxy using the full set of TE consensi generated by *RepeatModeler2*, as outlined in the previous steps of this Appendix. This step will generate the initial seed alignment for your TE family of interest, allowing you to skip **Step 2.2**.

This will output a new `.align` file which can be used below.

- Edit this to specify the name of the TE family to extract from the *RepeatMasker* alignment file (*e.g.* `rnd-3_family-103#LTR/Copia`).

- Note that, in this case, you will need to combine the TE ID (e.g. `rnd-X_family-Y`) with the classification information (e.g. `LTR/Copia`), separated by a pound sign.

`TE_family:` rnd-3_family-103#LTR/Copia

Show code

7.5 cont'd) Generate `.stk` file

This will output a new seed alignment file (`.stk` file), which can be used as input for **Step 2.3**.

- Edit this to specify the name of the genome fasta file (*e.g.* `dmel.fasta`).

`genome:` dmel.fasta

- Edit this to specify the species name (*e.g.* `Drosophila melanogaster`).

`taxon:` Drosophila melanogaster

Show code

7.6) **Alternative to Step 2.2** - Run **RepeatMasker** on Galaxy using single TE consensus and extract seed alignment from output

If you're having issues getting **Step 2.2** to complete in this notebook due to a large genome size, an alternative strategy is to run RepeatMasker on Galaxy using a filtered repeat library consisting of a single TE family. To do this:

1. Generate a TE library containing just your family of interest using **Step 2.1** in this notebook. Save this file and upload it to Galaxy along with your genome file.
2. Run RepeatMasker on Galaxy following the instructions in **Step 7.2**.
3. Upload the RepeatMasker alignment file from Galaxy to this notebook using **Step 7.2**.
4. Now, edit the details below and run this cell.

- Edit this to specify the name of the genome fasta file (*e.g.* `dmel.fasta`).

`genome:` dmel.fasta

- Edit this to specify the species name (*e.g. Drosophila melanogaster*).