Formatting MIRA output for GISAID upload

Bash scripts, ending with .sh should be executed in the following way: ./script.sh

Python scripts can be executed on the command line using the following command:

python my\_script.py (or python3 my\_script.py), check first if you have python installed on your command line. Alternatively scripts can be executed on the VS code platform (<https://code.visualstudio.com/> )

1. Run fastq files on the MIRA pipeline.
2. Unzip the Irma\_allconsensus\_bam.tar.gz file.
3. The pipeline will output results for each sample in subdirectories of the same name in the directory above. Each directory will contain fasta files, bam files, intermediate files and figures.
4. To determine the coverage across each of the eight segments, use qualimap (<http://qualimap.conesalab.org/> ).
5. To run quali map, use the ./Run\_IRMA\_qual\_full.sh script directory on the command line. First run edit the “WORKING\_PATH=” to the path where your individual result directories are. The script will produce a text file containing all id’s in your analysis and output from qualimap for all eight segments.
6. Compile all qualimap coverage results into a .tsv file using the extract\_qualimap\_metrics\_all\_collect co\_depth.py script. Update the path (collection\_dir) to where the qualimap results are located.
7. Eight csv files will be produced (cov\_results\_segment\_all.csv) plus the excel file containing the id’s.
8. Run the merge\_coverage\_files.py to merge all .csv files and the excel file into one spreadsheet (inspect the spreadsheet to familiarise yourself with its content).
9. Further format the spreadsheet to add the subtype and type column (formatted from the bam file column) using the edit\_merged\_spreadsheet.py script.
10. To upload data on GISAID, metadata for the sequences is required. So prepare that file separately and use the merge\_meta\_coverage\_one\_dataset.py script to merge the coverage spreadsheet with the metadata file. Make sure to edit the path and that the columns containing the sample id’s have the same heading of “sample”.
11. Format the merged spreadsheet from above with the extract\_meta\_for\_GISAID\_TEMPLATE\_FLU.py script. This will produce a spreadsheet ready for GISAID upload (make sure you convert to csv). Make sure to go through the script to change parameters that are specific to our lab. The script will also produce a text file with the sequence id’s.
12. Copy all consensus fasta files in one directory “amended\_consensus” using the copy\_IRMA\_all\_cons\_out.sh script.sh
13. Finally, use the copy\_IRMA\_GISAID\_out.sh script to gather all fasta files for GISAID.
14. Use the csv file from step 11 and the fasta file from step 12 to upload to GISAID using the CLI upload option (please check out the CLI page on the influenza CLI upload for more details).

* Obtain a client-ID by clicking the “Request client-ID” from where you downloaded the software.
* Upload using the following command:
  + ./fluCLI upload --username USERNAME --password PASSWORD --clientid CLIENT ID --log LOG --metadata template\_GISAID.csv --fasta GISAID.fasta --dateformat YYYYMMDD
* Log in to the GISAID website and inspect the submission before release. Look out for any warnings (e.g. frameshifts or missing gene detected), manually correct or discard any sequences not suitable for upload (if a sequence frameshifts, insertions or missing bases).