Genomic epidemiology tutorial

# Introduction

In this scenario, we have performed a high-level analysis using pangolin to see how connected viruses are. Now that there’s some transmission we can rule out, we want to dig into a couple of the relationships a little further.

Of concern are the sequences EDB005 and EDB010, a patient and a doctor whose sequences are part of the same lineage, so they might be connected in a transmission chain. To complicate matters, the doctor who provided sequence EDB010 recently met up with two friends from other parts of Scotland, who later tested positive for COVID-19 and were already sequenced as part of routine genomic surveillance. We want to investigate possible transmission chains between the doctor and their friends, as well as the patient. This could be important in determining the importance of within-country travel to transmission.

We are also interested in EDB001 and EDB006, who are on different wards, but are part of the same lineage. If they are part of the same transmission chain, the hospital may need to tighten up its infection prevention and control (IPC) measures.

# Installing civet and downloading data files

For this investigation, we’ll be using the tool civet, which can be found at <https://github.com/COG-UK/civet>, and documentation at https://artic-network.github.io/civet/.

Civet is a tool developed by Áine O’Toole, Verity Hill and other members of the Rambaut Lab in Edinburgh to perform rapid cluster investigations.

To install civet, run the following commands in a terminal window:

1. git clone https://github.com/COG-UK/civet.git and cd civet
2. conda env create -f environment.yml
3. conda activate civet
4. python setup.py install

This will clone the github repository that contains the civet code (step one), create and activate a conda environment where we can install packages (steps two and three) and then finally install civet in this environment (step four).

Next, ensure you have downloaded the files required to run this tutorial. These are the three background data files (background\_fasta.fasta, background\_metadata.csv and background\_tree.tree), the two input files (input\_metadata.csv and input\_sequences.fasta) and the config file (config.yaml). These files can be found at https://www.climb.ac.uk/artic-and-climb-big-data-joint-workshop/

Put them in the directory where you want to run civet. Please ensure there are no spaces in the path to this directory by typing pwd. If you can see spaces in the result, please move the files to a directory with no spaces, otherwise civet will not run.

Your directory file structure should look like this:



# Running civet

We will use the config.yaml to specify the options we’re interested in for this analysis. Civet is highly customisable, both in terms of its analytical pipeline and its final report (see <https://artic-network.github.io/civet/> for more details), but for now we will just use the options specified in the config file.

Type the following command into the command line:

civet -i config.yaml

And that’s it! You will see civet start to run by printing out its logo and associated information. This process will take approximately half an hour, depending on your computer’s processer. If you want to see more about what civet is doing, you can use the --verbose flag, which will print out additional information while civet is running.

# If you don’t want to run civet…

You don’t have to – the report can be found on the workshop website https://www.climb.ac.uk/artic-and-climb-big-data-joint-workshop/

# The civet report

The civet report begins with some summaries of the data, including which options were input. The tables are split into queries which were found in the background database (table 1) and those which had sequences provided by the user (table 2).

The report then moves onto phylogenies. Remember, civet cuts up the large background tree into local catchment trees. If sequences appear in different trees to each other, it means they are much less related to each other than those that are in the same tree. In this case, the trees have been coloured by the ward and whether the person was a patient or a health care worker.

Civet has also produced a timeline of dates that can help with determining whether cases are connected or not. As background, the COVID-19 incubation period is 2-14 days, and people can be infectious before they display symptoms. By combining this data with seeing which genomes are closely related to each other, it can make it easier to rule out transmission.

Below that, it’s also produced a snpit graph (https://github.com/aineniamh/snipit), which zooms in on the part of the genome containing the mutations on the branch leading up to the sequences of interest compared to the reference genome. This is useful if we want to check that it’s not missing or ambiguous data that’s leading to sequences not being identical; and to provide some more information about the underlying genomic data.

The snpit and the timeline are produced per tree, so as you scroll down you will see more groups of sequences represented. (NB the timeline is only produced if more than one sequence of interest is present in a given tree).

# Things to think about as you interpret the report:

1. What happened to sequences EDB003 and EDB004?
2. Are the doctor’s (EDB010) friends from Glasgow (730568) and Aberdeenshire (807866) are involved in the hospital outbreak? Why or why not?
3. Can we definitively say whether the doctor (EDB010) and the patient (EDB005) are connected? What information is there, and what more could be useful?
4. What can we say about EDB006 and EDB001?