Dear Dr Gonché,

I have now received the comments of two reviewers for your manuscript. They both find your work very interesting but point some important issues that need to be addressed, especially the lack of data test. Moreover, reviewer 1 had difficulty accessing the program.

Sincerely

Emmanuelle Lerat

Review by Bastien Boussau, 05 Mar 2023 16:22

Danesh et al. present a pipeline to select sequences according to a range of criteria from data downloaded from Gisaid. It can be used to select sequences from a specific range of dates, from specific regions, from specific sequencing laboratories, from specific viral lineages, by specifying the options in a yaml file. Further filtering options can be used as well according to a json file. Once the data has been filtered, the pipeline can align the sequences, construct and date a phylogeny, and build the configuration file for a BEAST2 birth-death skyline plot analysis. The manuscript describes the pipeline, and presents an example analysis that has been performed with the pipeline.

The manuscript is clear and easy to read. The tool provided is likely to be useful, is easy to install (although I report below one typo), with a very good documentation, but it is somewhat hard to test, because the authors have not provided a test data set. This is likely due to GISAID rules, which prevent distributing subsets of their data. However, the authors could build a mock data set, with a mock aligment (which can be very small), and mock metadata, on which their pipeline could run. This example might be useful to users who first want to try the tool on a small data set before they try it on their own.
Thank you for your suggestion. We added a test dataset with metadata and sequence data published on both GenBank and GISAID (because GISAID data is restricted to registered users).
=> A section in the Gitlab documentation now explains how to run this example.

Another recommendation I have would be to comment on the importance of the different priors users have to set for the parameters of the BDSKY analysis (in the minor comments below I point out that the "origin" parameter might be an important one). Some parameters may have a stronger influence than others, and their impact on the analysis may not be obvious to users. This information could be provided on their gitlab website.
This is a valid remark, which is common to any Bayesian analysis. The version of Beast2 we used is designed to provide default priors that minimise the risk of bias in the results. However, especially if the data is itself biased, results should always be handled with care.
=> We now mention this in the manuscript and in the documentation.

Minor comments:

l22: "were made available" : have been made available
l23: "This allowed" : has allowed
l31: "go from dates virus sequence data": dated
=> Thank you for the detailed feedback. These are now corrected in the manuscript.

l122: "This yields infectious periods varying from 1.2 to 36.5 days": perhaps specify the mean, and clarify a bit because the parameter was specified a couple of lines above as per

year, whereas this sentence is in days, which can be a bit confusing.
=> Thanks again. We now clarify this by explaining that the inverse of the end of infectious period rate is the average infectious period. We kept the units in years, and added the values in days in brackets.

l128: "The default prior for this parameter prior is a uniform distribution Uniform(0, 2) years.": this prior seems a bit dangerous for naive users who may be using the method in the future. If they don't change it, it seems like they would not be able to infer origin dates older than 2 years from the date of their analysis.
Yes, if the starting value or upper value of the origin parameter is lower than the maximal height of the tree, Beast will indeed return an error and not run the BDSKY analysis.
=> We modified the Cov-flow pipeline to add check steps for all the parameters, including checking that the condition of the starting value and the upper values are higher than the tree height. The default prior now is a Uniform distribution with a lower value of 0 and an upper value of max_height+2 years and a starting value of max_height+0.1 years.

Fig. 2 legend: "In panel c, the lined show" : lines
l155: "allow us to visualise": allows us
l161: "variant epidemic seems to occurred earlier and more frequently": have occurred
l168: "PACA experience a period of Delta variant growth": experienced
l180: "from the GISAID datased" : dataset
=> Thanks! These typoes are corrected in the manuscript.

Software test:
cd cov-flow : cd Cov-flow

Review by Gabriel Wallau, 30 Jan 2023 16:40

Danesh and collaborators presented COVFlow, a computational pipeline aimed to perform sample selection and phylodynamic analysis of SARS-CoV-2 sequences. Due to the huge amount of SARS-CoV-2 sequences available in public databases such pipelines are in much need to select datasets that are amenable to computational analysis and inferences. Therefore, COVFlow addresses an important bottleneck in the field of genomic surveillance particularly regarding the generation of virus transmission rate inferences that is a key information to inform the public health decision making process. However, from the application point of view this pipeline is able to perform similar steps already performed by other highly used software (i.e. Nextclade). In addition, I could not test the pipeline due to user permission restriction. In summary, I suggest a number of modifications and clarifications in the manuscript to be able to reassess its in more in detail.

We thank the reviewer for his careful reading and suggestion.

Comments and requests

Page 3 - line 31 - I suggest changing “dates virus sequence data” to “data stamped virus sequence data.”
Thank you for the suggestion!
=> We changed to “time-stamped virus sequence data”.

page 3 - lines 40-41. What authors meant with “However, these do not include a data filtration step based on metadata characteristics.”? The nextstrain CLI tool, which includes Augur in some steps, allows the user to filter data based on different metadata (see https://docs.nextstrain.org/projects/ncov/en/latest/guides/workflow-config-file.html), such as: collection date, pangolin lineage, genome length, host, geographic information (region, country, division, location). I suggest the authors clarify which metadata COVFlow can filter out that nextclade can not. Moreover, I recommend the authors to describe the advantages of each step of CovFlow (filtering, alignment, masking sites and build tree) when compared with nextstrain (https://docs.nextstrain.org/en/latest/learn/parts.html). From my point of view there are two new COVFlow features compared with nextstrain CLI, that is, subsampling appears to be a proportional sampling in the models (instead of a absolute number per sampling group model that can be set in nextclade) and the generation of XML file to be used on beast2.
We apologize for the unclear formulation. Nextstrain is indeed a very useful tool but there are a few noticeable differences with COVflow.
First, the JSON file in COVflow allows for more flexible data filtering. For example, it can select data if a column contains a certain word, allowing the user to filter data that may contain spelling mistakes or to select data from a group of laboratories that contain a common word (in our case CERBA) but don’t have the same names.
Second, as you noted, COVflow also allows the user to perform a subsampling in terms of the percentage of available data.
Third, COVflow follows the recommendations detailed here https://github.com/W-L/ProblematicSites_SARS-CoV2 to mask the sites, which are different in Nextstrain.
Fourth, and finally, it allows a phylodynamic inference.
=> We now more carefully describe the differences between COVflow and Nextstrain in the manuscript.

Page 6 - line 96 - Why the authors used the option “and ’addfragments’” if the sequences are almost all full length? Maybe the --add option is enough. Please clarify.
We initially used this option because it was recommended in the MAFFT documentation. However, you are correct in that it is not necessary since the sequences are mostly full length sequences.
=> We made the change in the manuscript and updated the code.

Page 7-9 - lines 137-152 - Regarding the selection of samples for BDSKY analysis, one key step is the selection of monophyletic clades and then performing Re estimates on them separately. Otherwise, Re estimates could be much biased by inferring transmission timing dynamics from unstable “clades”, which means that every run of the pipeline may generate a

different time-tree structure and reach different Re estimates. Did the authors include such a step on the pipeline? Please clarify.

You are correct in that we did not include this step in the pipeline. We also agree that this is an interesting suggestion.
To explore how the estimations vary (or not) for different runs and hence different trees, we ran 5 times the COVflow pipeline to obtain 5 different phylogenetic trees for the analysis with all the French data . For each of the 5 trees, we selected the two principal clades. For each subtree, a BDSKY analysis was performed with the same prior for each parameter. We estimated 9 varying R_e for each BDSKY analysis, with change-point times every month from 2021-06-01 to 2022-01-01. We found that the estimated R_e for each subtree was similar to the one estimated with the whole tree, and the estimates are similar for the 5 different tree sets.
=> These results are included in the manuscript with a new Figure 3.

Moreover, figure 2 lower section should be depicted with case numbers from each region and the whole country to evaluate if the Re estimates are compatible with the epidemiological curves. I suggest three different plots, one for each region considered.

Thank you for the suggestion! We now also show the reproduction number (R_e), which we inferred from the positivity rate of the screening tests performed in each region. We used the EpiEstim method to estimate R_e and the R code to generate the data is attached.

Interestingly, we see that there is sometimes a one-week shift between the peaks observed in the Beast inference and the ones observed in the incidence data. This is to be expected since incidence data is shifted compared to the state of the epidemic (by the average number of days between infection and screening). The phylodynamic inference is expected to be less biased in that respect.

=> We updated the figure with the Re from the incidence data.

At the moment of Delta variant spread the population had already a complex mix of acquired and vaccine induced immunity. It would be interesting to add the vaccination rate from each region through time in this figure as well.

The reviewer is correct in that vaccination did play a role in slowing down the spread of the Delta variant. However, the French context also played and the summer school holidays (in July and August) also most likely had a similar (it not larger) effect. Showing the vaccination on the figure would, therefore, potentially be misleading and probably make the figure too crowded.
=> We now discuss the factors that may have affected variations in R_e.

Page 9 - lines 168-171. Are there any other available genomic data that could provide some additional lines of evidence of a Delta growth at this time point besides inference tests?

Proportion of genomic defined lineages? One suggestion is to plot the lineage GISAID data itself from each region and France alongside Figure 2C.

France also performed variant-specific screening tests using PCRs and sequencing on a subset of these tests. We analysed such data earlier but, to keep the manuscript concise, only refer to these.
=> We now cite Alizon et al (2021, *Eurosurveillance*) and Sofonea et al. (2022, *Emerg Infect Dis*) to describe the types of SARS-CoV-2 lineage circulating over the time period considered.

Git Lab issues

Following the Gitllab instructions on installing COVflow, the git clone section returns: fatal: Could not read from remote repository.

We checked and, indeed, only the SSH cloning (and the direct download) were working. We are checking with the server IT helpdesk to address this issue.
=> We modified the gitlab documentation for the git clone command line.

The authors should clarify how to obtain the tsv metadata file. Can it be obtained from the general metadata present on the Download section of GISAID - EpiCov or it came from the metadata available after a sequence selection performed in the search interface of GISAID - EpiCov? If the metadata file has more columns than the ones specified on Metadata Fields would COVFlow still work?

As indicated in the documentation, COVflow would still work with only the two fields ‘strain’ or ‘date’, or with more columns than those indicated in the Metadata fields. In this case, it would still be possible to indicate how to filter on these new metadata fields using the JSON file.
=> We added a section explaining the different ways (custom selection, general download, *via* R package) to download the sequence data and metadata.

I suggest that the authors create a test dataset with fasta and metadata files or inform a way that the user can recover it from Gisaid and an associated step-by-step guide that could be followed by the user to perform a test analysis with the current json files present in examples directory. This will facilitate the user implementation of COVFlow through simple testing.

Thank you for the suggestion!
=> We added a test dataset (with data published on both GISAID and GenBank because of GISAID rules) and a section ‘Run an example’ in the documentation so the user can test the pipeline.