

**Supplementary table 1.** Benchmark for SARS-CoV-2 Africa Dashboard concurrent requests:-\*

Concurrence level	Complete requests	Failed requests	Requests per second (mean)	Time per request ** [ms] (mean)
10	5000	0	38.7	25.841
100	5000	0	225.45	4.436
500	5000	0	536.93	1.862
1000	5000	4	373.66	2.676

\* All the experiments were performed using the same wifi network

\*\*Across all concurrent requests.

## SUPPLEMENTARY INFORMATION

### **SARS-CoV-2 Africa Dashboard: An interactive tool for visualizing COVID-19 genomics data**

Joicymara S. Xavier<sup>1,2,3\*</sup>, Monika Moir<sup>1</sup>, Houriiyah Tegally<sup>1,7</sup>, Nikita Sitharam<sup>1</sup>, Wasim Abdool Karim<sup>1</sup>, James E. San<sup>1,7</sup>, Joana Linhares<sup>1</sup>, Eduan Wilkinson<sup>1</sup>, David B. Ascher<sup>4,5,10</sup>, Cheryl Baxter<sup>1,8</sup>, Douglas E. V. Pires<sup>4,5,6\*</sup>, Tulio de Oliveira<sup>1, 7, 8, 9\*</sup>

<sup>1</sup>Centre for Epidemic Response and Innovation (CERI), School of Data Science and Computational Thinking, Stellenbosch University, Stellenbosch, South Africa.

<sup>2</sup>Institute of Agricultural Sciences, Universidade Federal dos Vales do Jequitinhonha e Mucuri, Unaí, Brazil

<sup>3</sup>Instituto René Rachou, Fundação Oswaldo Cruz, Belo Horizonte, Brazil

<sup>4</sup>Systems and Computational Biology, Bio 21 Institute, University of Melbourne, Melbourne, Australia

<sup>5</sup>Computational Biology and Clinical Informatics, Baker Heart and Diabetes Institute, Melbourne, Australia

<sup>6</sup>School of Computing and Information Systems, University of Melbourne, Melbourne, Australia

<sup>7</sup>KwaZulu-Natal Research Innovation and Sequencing Platform (KRISP), Nelson R Mandela School of Medicine, University of KwaZulu-Natal, Durban, South Africa.

<sup>8</sup>Centre for the AIDS Programme of Research in South Africa (CAPRISA), Durban, South Africa

<sup>9</sup>Department of Global Health, University of Washington; Seattle, USA.

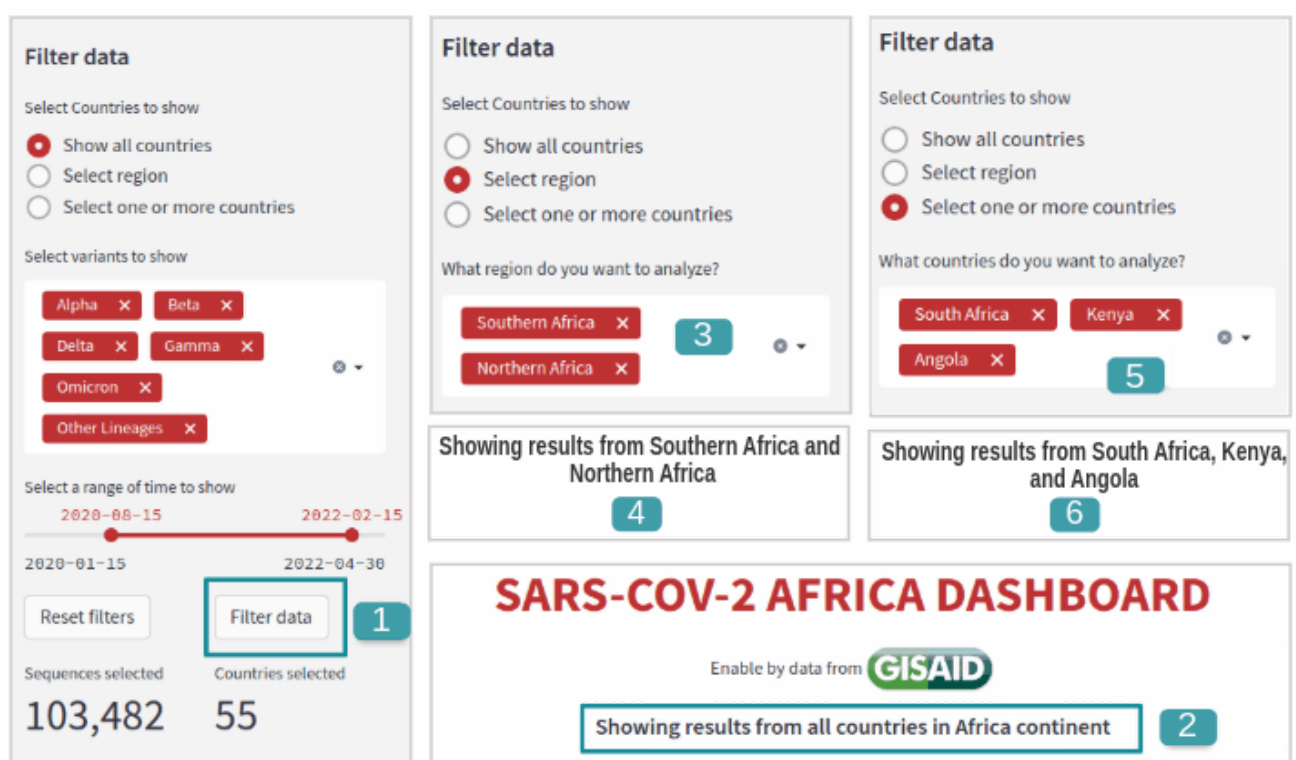
<sup>10</sup>School of Chemistry and Molecular Biosciences, University of Queensland

\*Corresponding authors: TdO, DEVP and JSX., email: [tulio@sun.ac.za](mailto:tulio@sun.ac.za); [douglas.pires@unimelb.edu.au](mailto:douglas.pires@unimelb.edu.au); [joicy.xavier@ufvjm.edu.br](mailto:joicy.xavier@ufvjm.edu.br), tel: +27 82 962 4219; +61 3 8344 8185; +55 38 991717950.

## HOW TO USE SARS-CoV-2 AFRICA DASHBOARD

SARS-CoV-2 Africa dashboard requires input data in two ways: by using a GISAID API or a custom tsv file formatted as per the provided template. To use an API, the developer is required to enter into a Data Provision Agreement with GISAID. The use of an API allows for automatic updating of the dashboard. Here we show two tutorials on how to use the SARS-CoV-2 Africa dashboard online and also how to reproduce the code locally.

### Tutorial 1. Using SARS-CoV-2 Africa Dashboard online: an example use case

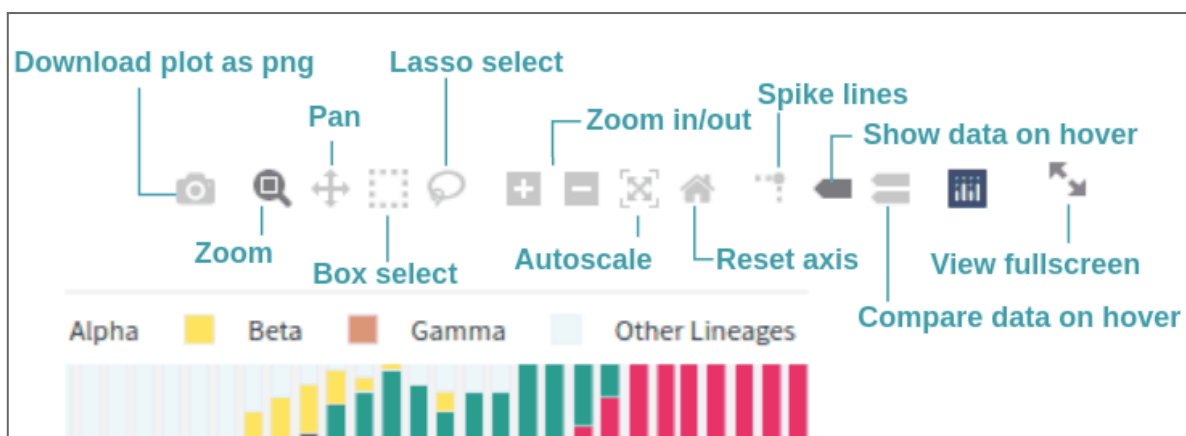


**Supplementary figure 1:** Use case example of filtering data selections to customize the SARS-CoV-2 variant, geographic and temporal data display on the SARS-CoV-2 Africa Dashboard.

When accessing the dashboard online, the default settings are to display the data for all African countries (Show all countries), for all SARS-CoV-2 variants (Select variants to show) with the full-time range of available data (Select a range of time to show). Filters may be applied with radio buttons and drop-down menus to customize the geographic distribution of data on the dashboard display. Similarly, a drop-down menu allows for specific variants to be chosen. A simple sliding time window may be used to adjust the temporal range of the display. These filters are applied with the

‘Filter data’ button as shown in panel [1]. When filters are implemented the number of sequences and countries selected, shown below the Reset filters and Filter data buttons, are updated. The dashboard display will also update to show which region or countries have been selected for data visualization [panel 2].

The ‘Select countries to show’ filter has three radio buttons to show all countries, select regions, and select one or more countries. When ‘select region’ is chosen, a drop-down menu becomes available with six African regions for selection. Multiple regions may be chosen [3] which will display in text on the dashboard [4]. When the ‘Select one or more countries’ radio button is selected, a drop-down menu with the list of African countries is displayed [5 and 6]. Any applied filters of region, countries, and variants may be removed by clicking on the exit button (X) of the particular filter or by pressing ‘Reset filters’ [1].



**Supplementary figure 2:** Several figure controls are available to download and modify the selected plot.

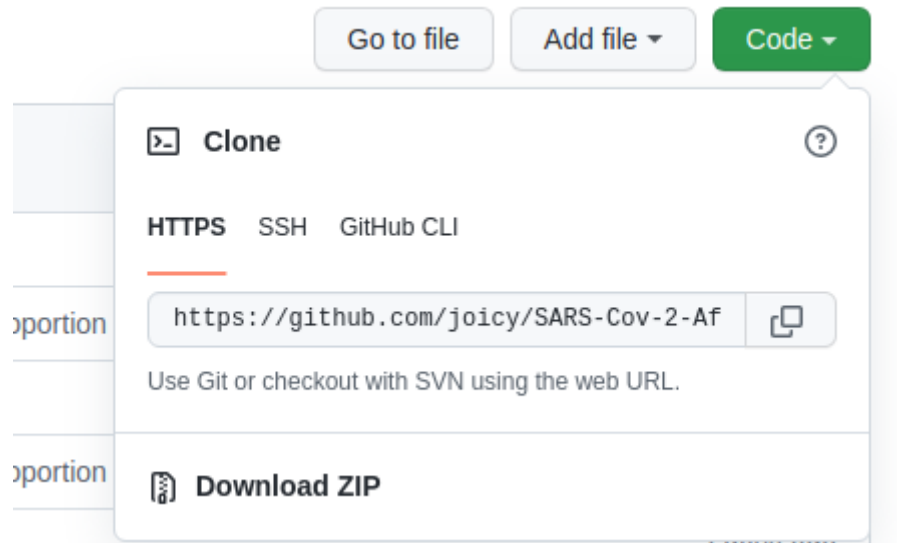
Figure controls are available on the top-right corner of each plot, their functions are as follows:

- Download the plot in png format
- The zoom control magnifies the area of the figure beneath the mouse cursor
- Pan around the figure
- Box select and lasso select controls may be used to highlight a particular area of the figure while the remaining portions of the figure are grayed out
- Zoom in (+) and zoom out (-)
- Autoscale automatically resizes the figure for a better visualization
- Reset axis resets the axes and figure size to the original display size
- Spike lines display a vertical and horizontal line from the mouse cursor location to the spike base on the x- and y- axes of the plot

- Show data on hover control enables the display of a single data label for the point directly beneath the mouse cursor
- Compare data on hover control allows for the display of multiple data labels for the point directly beneath the mouse cursor
- View fullscreen switches the active window to a full-screen view of the specific figure

## Tutorial 2. Using SARS-CoV-2 Africa Dashboard offline

1. Download the code at <https://github.com/joicy/SARS-Cov-2-Africa-dashboard>



2. Install the dependencies using Conda environment:

Start by opening your terminal and going into the project folder

```
(base) joicy@joicy-pc:~$ cd SARS-Cov-2-Africa-dashboard/  
(base) joicy@joicy-pc:~/SARS-Cov-2-Africa-dashboard$ conda env create -f requirements.yml
```

Type: `conda env create -f requirements.yml`

Activate the conda environment:

```
(base) joicy@joicy-pc:~/SARS-Cov-2-Africa-dashboard$ conda activate SARS-Cov-2-Africa-dashboard  
(SARS-Cov-2-Africa-dashboard) joicy@joicy-pc:~/SARS-Cov-2-Africa-dashboard$
```

Type: `conda (or source) activate SARS-Cov-2-Africa-dashboard`, the name of the environment will appear at the beginning of the line.

3. Now, you can run the dashboard using your own metadata or via setting up a GISAID API.

### Using metadata:

- Edit config.py file and set `data_source` variable with your option:  
`data_source="metadata"`

```
#Fill data_source variable with 'GISAID_API' if you are going to use GISAID feed data or 'metadata' to use a metadata file
data_source = "metadata"
```

- Create your metadata based on `data/template_metadata.csv` (required columns shown below) and save it in `data/metadata.csv`

	A	B	C	D	E	F	
1	lineage	collection_date	subm_date	region	country	province	
2	B.1.351	2021-03-18	2022-02-21	Africa	Kenya	Nairobi	
3	B.1.612	2020-12-01	2021-06-24	Africa	Gabon		
4	B.1.351	2020-12-01	2021-06-24	Africa	Gabon		
5	B.1.612	2021-01-01	2021-06-24	Africa	Gabon		
6	B.1.351	2021-01-01	2021-06-24	Africa	Gabon		
7							
8							

### Using GISAID API:

- Edit config.py and set `data_source` variable with your option:  
`data_source="GISAID_API"`
- Work with GISAID to get a Data Provision Agreement.
- Define the following environment variables in config.py:
  - GISAID\_URL
  - GISAID\_USERNAME
  - GISAID\_PASSWORD

```
#Fill data_source variable with 'GISAID_API' if you are going to use GISAID feed data or 'metadata' to use a metadata file
data_source = "GISAID_API"

# Setup GISAID variables if you are using GISAID feed data
GISAID_URL = 'https://www.gisaid.org/provision.json.xz'
GISAID_USERNAME = 
GISAID_PASSWORD = 
```

- Perform the edits required to customize your data if your needs differ from the standard in `source/data_process.py`
- If your data is not from the African continent, you must replace the geojson file in `data/africa.geojson`. We recommend [this repository](#) for accessing geojson files.