Supplementary Information for

CovRadar: Continuously tracking and filtering SARS-CoV-2 mutations for molecular surveillance

Alice Wittig, Fábio Miranda, Ming Tang, Martin Hölzer, Bernhard Y. Renard and Stephan Fuchs

Stephan Fuchs.
E-mail: FuchsS@rki.de

This PDF file includes:

Supplementary text
Figs. S1 to S4
Tables S1 to S2
SI References

Supporting Information Text

1. Pipeline workflow

This section explains in detail the algorithms and third party tools used in the analysis workflow used to generate the results displayed on CovRadar's website as well as the local PDF reports. An overview of this workflow is shown in Fig. S3, while further details of the algorithms are described in the following subsections.

A. Files extraction. CovRadar accepts as input compressed FASTA files containing the genomes and TSV files with their respective metadata. The file extraction is automated by a shell script capable of recognizing and extracting multiple formats, namely: 7z (.7z), bzip2 (.bz2), gzip (.gz), RAR (.rar), TAR (.tar), TBZ2 (.tar.bz2 or .tbz2), TGZ (.tar.gz or .tgz), Z (.Z) and zip (.zip).

B. Datasets merging. CovRadar's pipeline accepts multiple different sources as input. For example, it is possible to simultaneously use EBI, Charité and other third party datasets. This can be beneficial if those different sources have complementary data. The datasets merging is performed by a Python script that accepts as input only DNA sequences with IUPAC nucleotide codes (1), as shown on Table S1. Any sequences with unrecognized characters are removed to avoid errors in further steps of the workflow. File handling is leveraged by Biopython (2) and pandas (3) for the FASTA and TSV files, respectively.

C. Spikes extraction. In order to extract the spikes, first the genomes are aligned to the spike of Wuhan-Hu-1 (NC_045512.2), the first case from Wuhan. This alignment is performed by pblat (4), using default parameters. The software pblat is a scalable implementation of the popular aligner blat, which is especially useful in cases where other aligners may have trouble, e.g., when sequences are too long or the alignments have large gaps (4). Afterwards, an in-house Python script extracts the spike sequences from the genomes according to the coordinates detected by pblat, keeping only alignments with an N content smaller than or equal to 5% and a length deviation also smaller than or equal to 5% when compared to the spike from Wuhan-Hu-1. Filtering the spikes this way allows for an easy exclusion of sequences with low quality or with assembly errors. This script also employs pandas (3) and Biopython (2) to manipulate the TSV and FASTA files.

D. Multiple sequence alignment. The codon-aware multiple sequence alignment (MSA) of the extracted spike sequences is performed by VIRULIGN (5). The spike from Wuhan-Hu-1 is used as a reference and the results are exported as a global alignment composed of nucleotides.

E. Computation of consensus sequences. Before the consensus sequences are computed, first the spike from Wuhan-Hu-1 is extracted from the MSA produced by VIRULIGN, in order to be kept as reference. Afterwards, the MSA's sequences are divided by country and calendar weeks to allow an easier interpretation of the results. A list of selected countries used to compute the results can be found on Table S2.

The calendar weeks are computed using the ISO standard provided by the Python library isoweek, while sequences with incomplete or invalid dates are removed to keep the integrity of the results. Finally, the consensus sequences are computed from those divided subsets. The consensus algorithm counts how many A, T, C, G or gaps (-) appear per position, keeping whichever has the most occurrences or alphabetical order in case of ties. If none of those options arise, then an N is inserted in that position, thus accounting for ambiguous base pairs.

F. Numbering. To facilitate comparative results, we have introduced a numbering system based on Wuhan-Hu-1. With this numbering each MSA position can be converted to the corresponding position of the first case. The script takes the aligned first case. Since the unaligned first case has no gaps, an alignment will at most result in a longer sequence. Each gap corresponds to one insertion. For deletions the MSA will contain the gaps, but the length of the first case will not change. To get the corresponding positions of Wuhan-Hu-1, the script increments the position at every base that is not a gap. If there is a gap, the last position is taken instead with a suffix ".X", where X stands for an integer. The result is stored in a TSV table.

G. Variant counting. For the VCF file generation, the MSA and the aligned Wuhan-Hu-1 sequence are required as input. The first case will be the reference column (REF) in the VCF file. The script goes through every position in the MSA and adds differences as alternative alleles (ALT). The position refers to the MSA and is stored in the column POS. Only positions with variations are stored in the VCF file. Due to performance reasons, first, each row of the VCF file is saved in a temporarily directory. After each row is completed, they get merged into one VCF file.

H. Basic statistics. Basic statistics are computed with BCFtools (6, 7) and RAxML-NG (8). For BCFtools the stats parameter is used, while for RAxML-NG we use the parameter --check. As a best-fitting nucleotide substitution model and evolutionary model, a generalized time-reversible plus gamma distribution (+G) was chosen by ModelTest-NG (9). We specified to optimize the base frequencies by maximum-likelihood (+FO).

I. Plots creation. The PDF reports contain two charts, one representing the nucleotides substitutions and the other showing the allele frequency. The substitutions are computed by BCFtools, which stores its results in text format. Those results are parsed with the Python library *re* using regular expressions, then they are plotted with the library *matplotlib*. The PDF reports use a modified allele frequency algorithm of the web application to create the allele frequency plot. Please, see Section 3.B for a detailed explanation and the differences.

2. MySQL Database

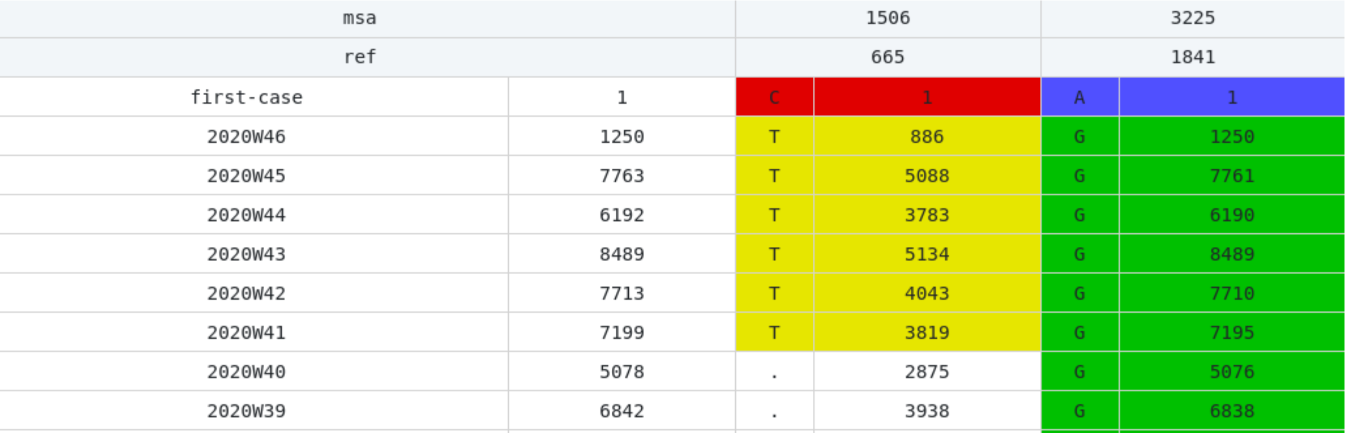
The results of our analysis are stored in a MySQL database that the app has access to. It contains following seven tables (Fig. S4):

- *global_metadata* contains the input metadata that comes with the sequences.
- *global_id* contains the samples from the metadata with consecutive integers as unique IDs.
- *global_statistics* contains the content of the log files that were created during the alignment.
- *consensus* contains the consensus sequences of each country and calendar week, as well as the number of sequences from which the consensus sequence was built.
- *base_count* contains for each of the consensus sequences and positions the coverage of the sequences that have the consensus base.
- *global_position* contains for each MSA position the corresponding spike position in Wuhan-Hu-1.
- *global_SNP* contains the alternative alleles of each sequence for each MSA position regarding Wuhan-Hu-1.

3. Web Application

A. Implementation Details. The web application of CovRadar is written in Flask (https://github.com/pallets/flask), a web framework in Python, and Dash(https://github.com/plotly/dash), that is built on top of Flask, React.js (https://github.com/facebook/react) and Plotly.js (10). This enables easy to integrate interactive user interfaces keeping Python syntax for the backend logic and algorithms. CovRadar runs on de.NBI cloud on a Gunicorn webserver (https://github.com/benoitc/gunicorn) behind Nginx (https://www.nginx.com/) as reverse proxy. The local version can be started with the integrated Flask server or with the provided Docker container (https://docs.docker.com/). The code for each page is modularized, which allows to create and include them independently. This makes it easy to add new pages with new tables and plots. CSS selectors can be used to choose between portrait and landscape format. Detailed instructions for installation and adding custom pages can be found on https://gitlab.com/dacs-hpi/covradar.

B. Allele frequency plot. The allele frequency plot shows the frequency of alternative alleles in relation to Wuhan-Hu-1 or a selected consensus sequence (Fig. S2) For the frequencies of the alternative alleles of a requested subsample, these sequences are filtered out of the VCF file using country, start date, end date, origin and host. The reference in the VCF file is Wuhan-Hu-1. Thus, if the user has selected a consensus sequence against which the frequencies should be calculated, this must be taken into account. For this purpose the positions that show differences between the consensus sequence and Wuhan-Hu-1 are determined and it is checked if they show alternative alleles in the VCF file. If not, all sequences for this position have the base of the first case. This means that in relation to the consensus base at this position the frequency of the alternative alleles is 1. For all other positions, where the first case base and the consensus differ, the alternative alleles are identified again. For some cases no clear statement can be made if an alternative allele is present because in contrast to Wuhan-Hu-1 and the consensus sequences degenerated bases can occur in the sequences. For example, if there is only the information that the sequence base is a purine and the reference base is adenine or guanine. In this case only pyrimidines are counted as alternative alleles and otherwise excluded. Finally, for each position the number of non-excluded alternative bases is divided by the number of non-excluded sequences at that position and the result is returned as a list of frequencies. Since only non-degenerated bases as well as gaps were used in the consensus sequence, it may occur that there is no coverage for a position. In this case the consensus sequence shows the base N. If the consensus base is N, no frequency is determined. For the PDF report the reference sequence is Wuhan-Hu-1 and whole dataset is taken.



msa		1506		3225	
ref		665		1841	
first-case	1	C	1	A	1
2020W46	1250	T	886	G	1250
2020W45	7763	T	5088	G	7761
2020W44	6192	T	3783	G	6190
2020W43	8489	T	5134	G	8489
2020W42	7713	T	4043	G	7710
2020W41	7199	T	3819	G	7195
2020W40	5078	.	2875	G	5076
2020W39	6842	.	3938	G	6838

Fig. S1. Consensus table of the global dataset showing the occurred mutations at codon position 222 (nucleotide spike position 665 regarding to Wuhan-Hu-1) and codon position 614 (nucleotide spike position 1841). The table contains position, consensus base and coverage of changes in the consensus base regarding to Wuhan-Hu-1. The table header shows the position with respect to the multiple sequence alignment (msa) or with respect to Wuhan-Hu-1 (ref). The row starting with "first-case" contains the corresponding bases of the first case (Wuhan-Hu-1). Below are the consensus bases per calendar week. If they match the first case, they are marked with a dot. Next to the calendar week is the sequence coverage for that week. Next to the bases is the number of sequences that have this base at this position.

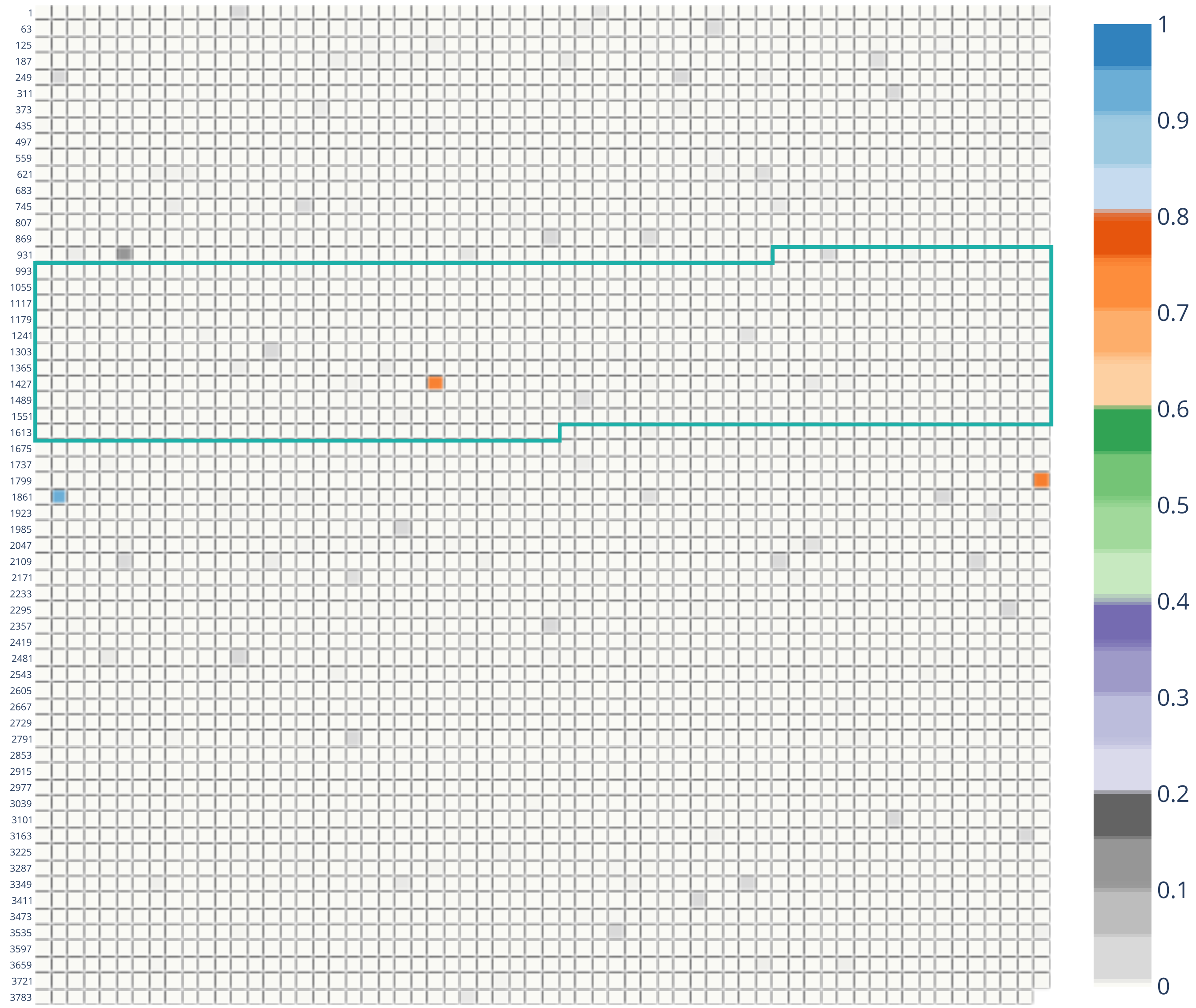


Fig. S2. Allele frequency plot of the Australian sequences with framed receptor binding domain (RBD). Each block represents a nucleotide in the MSA of spike gene sequences. Coordinates on the left are related to the MSA position. It shows frequencies > 0.7 for codon positions 477 and 613.

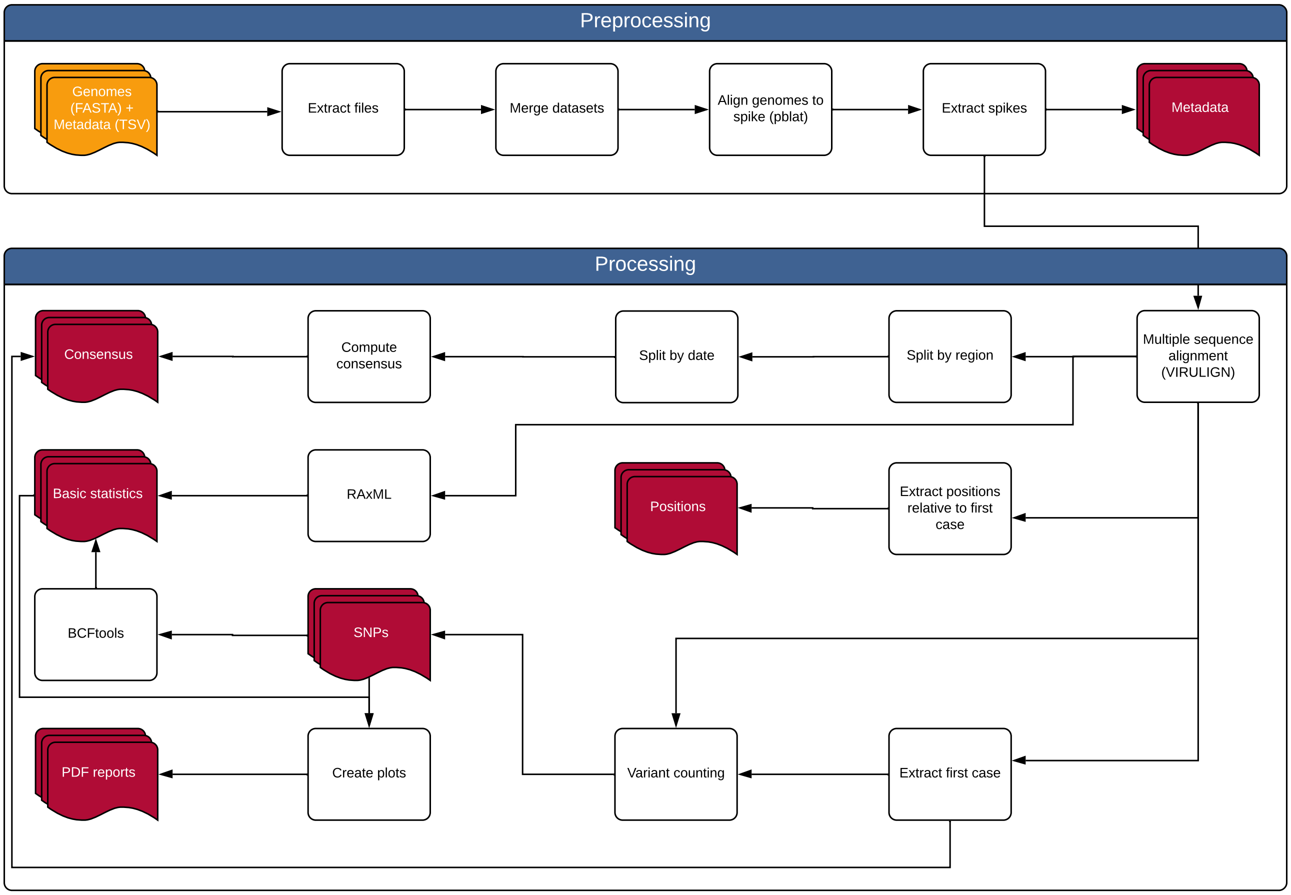


Fig. S3. Analysis workflow used to generate the results displayed on CovRadar's website. Orange blocks depict input files, while white blocks are processing steps of the pipeline and red blocks represent output that is displayed on the website. The pipeline accepts as input compressed FASTA files containing the sequences and TSV files with their metadata. One or more different data sources can be used simultaneously, e.g., Charité and EBI. First the pipeline extracts the files, then merges the datasets if more than one is used. Afterwards, pblat is used to align the input genomes against the spike sequence from Wuhan-Hu-1. Then the spikes are extracted with the coordinates reported by pblat. Next, VIRULIGN is used to perform a codon aware multiple sequence alignment (MSA) of the extracted spikes. Before computing the consensus sequences from the MSA, the sequences are first separated by country and calendar week. Additionally, the Wuhan-Hu-1 is extracted from the MSA and added to the consensus sequences to be used as reference. The coordinates relative to Wuhan-Hu-1 are extracted from the MSA with an in-house script. Finally, the variant counting is performed by an in-house script and basic statistics are computed with RAxML and BCFtools.

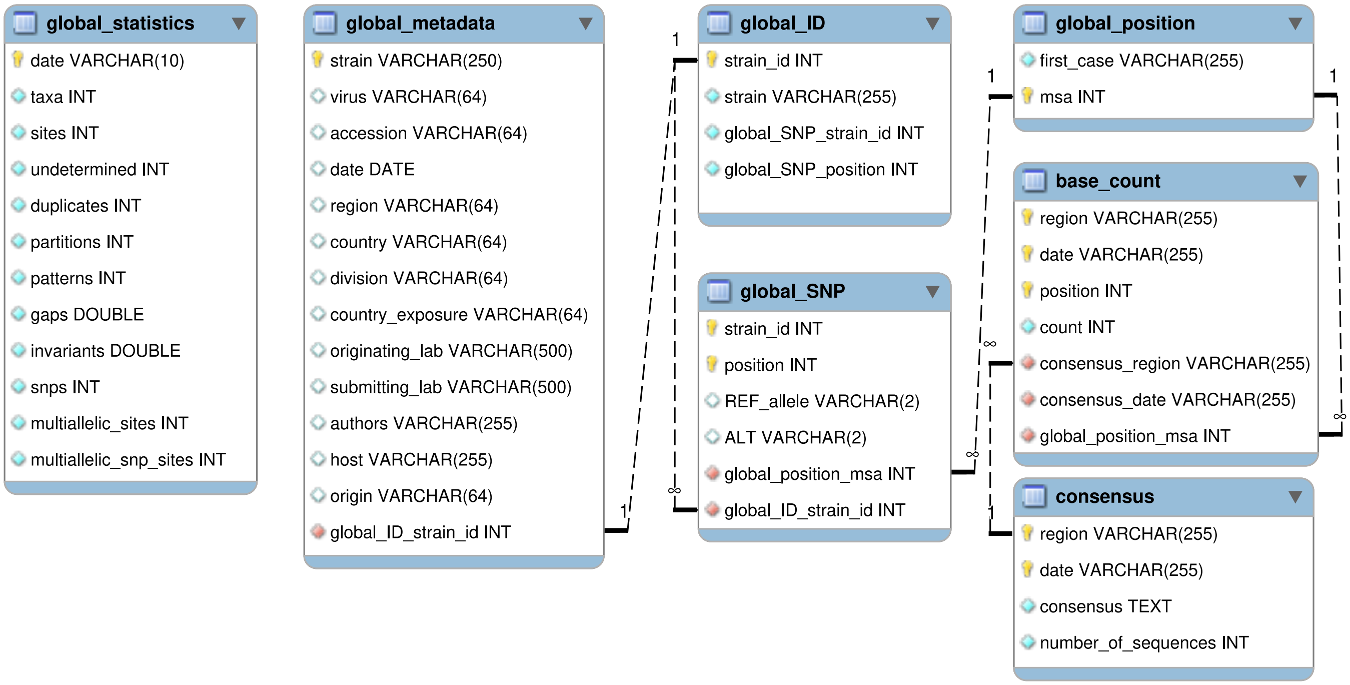


Fig. S4. Enhanced entity-relationship (EER) diagram of the MySQL database. Relations are represented as dashed lines where 1 and ∞ show one-to-one or one-to-many relationships. Primary keys are indicated with keys, foreign keys with red colored symbols, filled diamonds are NOT NULL attributes and empty diamonds can be NULL.

Table S1. IUPAC nucleotide codes accepted by the merging algorithm. Sequences with different characters are automatically removed to avoid errors in further steps of the pipeline.

Recognized characters										
A	T	C	G	N	Y	R	W	S	K	M
D	V	H	B	-	a	t	c	g	n	
y	r	w	s	k	m	d	v	h	b	

Table S2. List of countries selected to divide the dataset.

Region						
Global (all countries)	Canada	Finland	Israel	Moldova	Poland	Suriname
Algeria	Chile	France	Italy	Mongolia	Portugal	Sweden
Argentina	China	Gambia	Jamaica	Montenegro	Qatar	Switzerland
Australia	Colombia	Georgia	Japan	Morocco	Romania	Taiwan
Austria	Costa Rica	Germany	Jordan	Myanmar	Russia	Thailand
Bahrain	Crimea	Ghana	Kazakhstan	Nepal	Saudi Arabia	Tunisia
Bangladesh	Croatia	Greece	Kenya	Netherlands	Senegal	Turkey
Belarus	Cuba	Guam	Kuwait	New Zealand	Serbia	Uganda
Belgium	Cyprus	Guatemala	Latvia	Nigeria	Sierra Leone	Ukraine
Belize	Czech Republic	Hong Kong	Lebanon	North Macedonia	Singapore	United Arab Emirates
Benin	Democratic Republic of the Congo	Hungary	Lithuania	Norway	Slovakia	United Kingdom
Bosnia and Herzegovina	Denmark	Iceland	Luxembourg	Oman	Slovenia	Uruguay
Brazil	Dominican Republic	India	Madagascar	Pakistan	South Africa	USA
Brunei	Ecuador	Indonesia	Malaysia	Panama	South Korea	Venezuela
Bulgaria	Egypt	Iran	Mali	Peru	Spain	Vietnam
Cambodia	Estonia	Ireland	Mexico	Philippines	Sri Lanka	Zambia

References

1. AD Johnson, An extended iupac nomenclature code for polymorphic nucleic acids. *Bioinformatics* **26**, 1386–1389 (2010).
2. PJ Cock, et al., Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25**, 1422–1423 (2009).
3. W McKinney, , et al., pandas: a foundational python library for data analysis and statistics. *Python for High Perform. Sci. Comput.* **14** (2011).
4. M Wang, L Kong, pblat: a multithread blat algorithm speeding up aligning sequences to genomes. *BMC bioinformatics* **20**, 28 (2019).
5. PJ Libin, K Deforche, AB Abecasis, K Theys, Virulign: fast codon-correct alignment and annotation of viral genomes. *Bioinformatics* **35**, 1763–1765 (2019).
6. V Narasimhan, et al., Bcftools/roh: a hidden markov model approach for detecting autozygosity from next-generation sequencing data. *Bioinformatics* **32**, 1749–1751 (2016).
7. P Danecek, SA McCarthy, Bcftools/csq: haplotype-aware variant consequences. *Bioinformatics* **33**, 2037–2039 (2017).
8. AM Kozlov, D Darriba, T Flouri, B Morel, A Stamatakis, Raxml-ng: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. *Bioinformatics* **35**, 4453–4455 (2019).
9. D Darriba, et al., Modeltest-ng: a new and scalable tool for the selection of dna and protein evolutionary models. *Mol. biology evolution* **37**, 291–294 (2020).
10. Plotly Technologies Inc., Collaborative data science (https://plot.ly) (2015).