aPhyloGeo-Covid: A Web Interface for Reproducible Phylogeographic Analysis of SARS-CoV-2 Variation using Neo4j and Snakemake

Wanlin Li[‡*], Nadia Tahiri[‡]

Abstract—The gene sequencing data, along with the associated lineage tracing and research data generated throughout the Coronavirus disease 2019 (COVID-19) pandemic, constitute invaluable resources that profoundly empower phylogeography research. To optimize the utilization of these resources, we have developed an interactive analysis platform called aPhyloGeo-Covid, leveraging the capabilities of Neo4j, Snakemake, and Python. This platform enables researchers to explore and visualize diverse data sources specifically relevant to SARS-CoV-2 for phylogeographic analysis. The integrated Neo4j database acts as a comprehensive repository, consolidating COVID-19 pandemic-related sequences information, climate data, and demographic data obtained from public databases, facilitating efficient filtering and organization of input data for phylogeographical studies. Presently, the database encompasses over 113,774 nodes and 194,381 relationships. Additionally, aPhyloGeo-Covid provides a scalable and reproducible phylogeographic workflow for investigating the intricate relationship between geographic features and the patterns of variation in diverse SARS-CoV-2 variants. The code repository of platform is publicly accessible on GitHub (https://github.com/tahiri-lab/iPhyloGeo/tree/iPhylooGeo-neo4j), providing researchers with a valuable tool to analyze and explore the intricate dynamics of SARS-CoV-2 within a phylogeographic context.

Index Terms—Phylogeography, Neo4j, Snakemake, Dash, SARS-CoV-2

Introduction

Phylogeography is a field of study that investigates the geographic distribution of genetic lineages within a particular species, including viruses. It combines principles from evolutionary biology and biogeography to understand how genetic variation is distributed across various spatial scales [1]. In the context of viruses, phylogeography aims to uncover the evolutionary history and spread of viral lineages by analyzing their genetic sequences and geographical locations. By examining the genetic diversity of viruses collected from various geographic locations, researchers can reconstruct the patterns of viral dispersal and track the movement and transmission dynamics of viral populations over time [2] [3] [4]. In phylogeographic studies of viruses, the integration of genetic sequences, geographic information, and temporal data is essential. Integrating genetic sequences with geographical data enables researchers to conduct robust analysis of phylogenetic relationships among viral strains and uncover intricate patterns of viral migration and transmission across diverse regions.Through the integration of genetic and temporal information, researchers can derive insights into the timescale of viral evolution and elucidate the origins as well as dispersal patterns of distinct viral lineages [5].

Throughout the COVID-19 pandemic, researchers worldwide sequenced the genomes of thousands of SARS-CoV-2 viruses. These endeavors have significantly enhanced researchers' ability to analyze the intricate temporal and geographic dynamics of virus evolution and dissemination, consequently playing a pivotal role in informing the development of effective public health strategies for the proactive control of future outbreaks. However, the abundance of genetic sequences and the accompanying geographic and temporal data are scattered across multiple databases, making it challenging to extract, validate, and integrate the information. For instance, in order to conduct a phylogeographic study in SARS-CoV-2, a researcher require access to data regarding the geographic distribution of specific lineages. This includes information on the predominant countries in which these lineages are prevalent, along with the earliest and latest recorded detection dates. The Pango Lineages Report serves as a valuable resource for obtaining such data [6]. Following this, researchers can utilize databases such as NCBI Virus resource [7] or GISAID [8] to access sequencing data corresponding to the identified country and lineage. Daily climate data (e.g., humidity, wind speed, and temperature) for each location involved during the pandemic can be obtained from reputable sources such as NASA/POWER DailyGridded Weather [9]. To supplement the analysis, epidemiological information, including COVID-19 testing and vaccination rates, can be sourced from projects such as Our World in Data [10]. In summary, conducting phylogeographic research in viruses entails not only the meticulous screening and selection of sequencing data but also the proficient management of associated geographic information and the integration of substantial volumes of environmental data. This multifaceted process can be time-consuming and susceptible to errors. The challenges associated with data collection, extraction, and integration have hindered the advancement of phylogeographic research within the field [11] [12].

To tackle these challenges, we employed the highly scalable and adaptable Neo4j graph database management system [13]

* *Corresponding author: Nadia.Tahiri@USherbrooke.ca*
‡ *Department of Computer Science, University of Sherbrooke, Sherbrooke, Canada*

for the storage, management, and querying of extensive SARS-CoV-2 variants-related data. Differing from traditional relational databases that employ tables and rows, Neo4j represents data as an interconnected network of nodes and relationships [14]. Graph theory, with its inherent advantages in relation analysis, has found extensive applications in Phylogeny. For instance, Laddada et al. (2022) [15] employed Neo4j to track and analyze mutation occurrences by treating each nucleotide site of SARS-CoV-2 sequences as a node, thereby exploring the connections between mutations. In our research, we utilize graph theory to trace the relationships among location, environmental factors, and lineages. By leveraging graph theory, this framework offers a robust foundation for modeling, storing, and analyzing intricate relationships between entities [16] [17].

On the other hand, while recent phylogeographic studies have extensively analyzed the genetic data of species across different geographic regions, many have primarily focused on species distribution or provided visual representations, without investigating the correlation between specific genes (or gene segments) and environmental factors [18] [19] [20] [21]. To bridge this gap, a novel algorithm applying sliding windows to scan the genetic sequence information related to their climatic conditions was developed by our team [22]. This algorithm utilizes sliding windows to scan genetic sequence information in relation to climatic conditions. Multiple sequences are aligned and segmented into numerous alignment windows based on predefined window size and step size. To assess the relationship between variation patterns within species and geographic features, the Robinson and Foulds metric [23] was employed to quantify the dissimilarity between the phylogenetic tree of each window and the topological tree of geographic features. Nonetheless, this process was computationally intensive as each window needed to be processed independently. Additionally, determining the optimal sliding window size and step size often required multiple parameter settings to optimize the analysis. Thus, reproducibility played a critical role in this process.

To address these challenges, we devised a phylogeographic pipeline that harnesses the capabilities of Snakemake, a modern computational workflow management system [24]. Distinguishing itself from other workflow management systems such as Galaxy [25] and Nextflow [26], Snakemake stands out as a Python-based solution, guaranteeing exceptional portability and the convenience of executing Snakefiles with a Python installation [27]. Leveraging various Python packages, including Biopython [28] and Pandas [29] [30], the Snakemake workflow efficiently handles tasks such as sequencing data reading and writing, as well as conducting phylogenetic analysis. Given these capabilities, Snakemake serves as an optimal choice for aPhyloGeo-Covid. Furthermore, Snakemake supports parallel execution of jobs, significantly enhancing the performance and speed of the pipeline. This pipeline implementation facilitates efficient and reproducible analysis, thereby streamlining the phylogeographic research workflow of the aPhyloGeo-Covid.

With a clear focus on addressing the aforementioned limitations, this study aims to develop an integrated, open-source phylogeographic analysis platform. This platform consists of two vital components: data pre-processing and phylogeographical analysis. In the data pre-processing phase, we employ searchable graph databases, enabling rapid exploration and offering a visual overview of SARS-CoV-2 lineages and their associated environmental factors. This efficient approach allows researchers to navigate through vast datasets and extract pertinent information for their analyses. In the subsequent phylogeographical analysis phase, our modularized Snakemake workflow is utilized to examine how genetic variation patterns within different SARS-CoV-2 variants align with geographic features. Leveraging this workflow, researchers can systematically and reproducibly investigate the relationship between viral genetic diversity and specific geographic factors. By adopting this comprehensive approach, a deeper understanding of the intricate interplay among viral evolution, transmission dynamics, and environmental influences can be achieved.

Methodology

A diverse range of data sources pertaining to SARS-CoV-2, covering the period from January 1, 2020, to December 31, 2022, were meticulously extracted, transformed, and loaded into a Neo4j graph database. These sources encompassed:

(1) SARS-CoV-2 sequences from the SARS-CoV-2 Data Hub [31]
(2) Lineage development information from Cov-Lineages [6]
(3) Population density by country, positivity rates, vaccination rates, diabetes rates, aging data from Our World in Data [10]
(4) Climate data from NASA/POWER [9]

To enable efficient querying, configuration of analysis parameters, and output generation within the database, a driver object was established using the Neo4j Python driver to establish seamless connections with the Neo4j database. For phylogeographic analysis, a streamlined workflow was implemented using the Snakemake workflow management system, ensuring an efficient and structured analysis process. Moreover, the interactive visualization capabilities offered by the Dash-Plotly library [32] [33] were leveraged for data exploration, analysis parameter setting, and interactive visualization of results, enhancing the interpretability and user-friendliness of the platform.

Data Integration

Within the Neo4j database, five labels were employed to effectively organize the data, encompassing Lineage, Protein, Nucleotide, Location, and Location Day (See Figure 1). The Protein and Nucleotide labels serve as repositories for sequencing data information, including accession number, sequence length, collection date, and collected country. The Lineage label stores lineage development information, encompassing the most common country, latest date, and earliest date associated with each lineage. Climate information such as temperature, precipitation, wind speed, humidity, and sky shortwave irradiance for each location and specific day is stored under the LocationDay label. The Location label contains fundamental information regarding hospitals, health, and the economy of each country, encompassing GDP, median age, life expectancy, population, proportion of people aged 65 and older, proportion of smokers, proportion of extreme poverty, diabetes prevalence, human development index, and other pertinent factors (See Table 1).

Lineage nodes establish connections with Nucleotide and Protein nodes, representing the relationships between lineages and their corresponding genetic sequence data. Moreover, Lineage nodes establish relationships with Location nodes, utilizing the most common occurrence rate as a property. This design empowers researchers to determine the most common countries based

TABLE 1: *Neo4j schema labels and properties for data integration.*

Label	Properties List
Protein	accession number, sequence length, collection date, collected country
Nucleotide	accession number, sequence length, collection date, collected country
Lineage	most common country, latest date, earliest date
Location Day	temperature, precipitation, wind speed, humidity, sky shortwave irradiance
Location	GDP, median age, life expectancy, population, proportion of people aged 65 and older, proportion of smokers, proportion of extreme poverty, diabetes prevalence, human development index

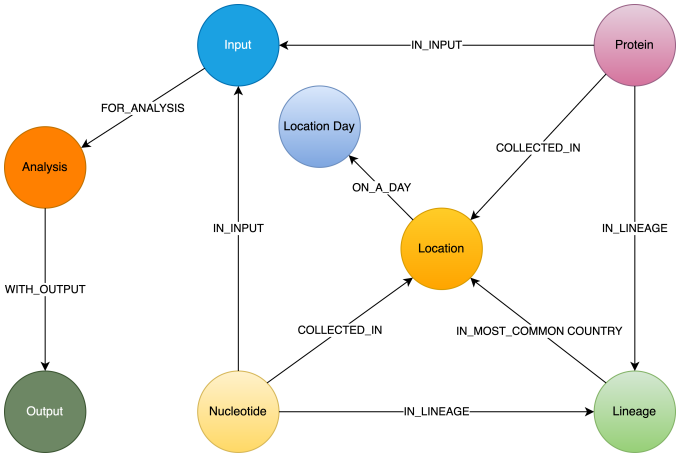

Fig. 1: *Schema of Neo4j Database for Phylogeographic Analysis of SARS-CoV-2 Variation. The schema includes key entities and relationships essential for organizing and querying data related to samples of protein, samples of nucleotide, locations, lineages, analysis input, output and parameters. Each entity represents a distinct aspect of the analysis process and facilitates efficient data organization and retrieval.*

on lineage names or search for lineages that were predominant in specific countries during specific time periods. This well-structured and interconnected design within the Neo4j database enhances the ability to explore, analyze, and extract meaningful insights from the integrated phylogeographic dataset.

Input exploration

An interactive platform using Dash-Plotly [32] [33] was developed for efficient data exploration and selection. The integration of the Dash platform with the Neo4j graph database allows for the seamless retrieval of pertinent data from interconnected nodes based on user-provided keywords related to lineages or locations. This functionality enables efficient identification and filtering of datasets for subsequent phylogeographic analysis. The integration of the powerful Neo4j database with the user-friendly interactive platform facilitates seamless data exploration and selection, supporting researchers in their comprehensive analysis of SARS-CoV-2 variation.

The aPhyloGeo-Covid offers two distinct approaches for selecting input datasets: 1) lineage-based approach for retrieving corresponding sequences based on selected lineage name and 2) location-based approach for retrieving corresponding sequences based on selected location and time period.

1. Lineage-based approach for retrieving corresponding sequences based on selected lineage name

The multi-step process is facilitated by the Neo4j Python package [34] and the interactive Dash web page. Initially, specific lineages of interest are selected from a checklist provided on the Dash web page. Subsequently, the selected lineages are utilized to query the graph database, extracting information about the predominant countries where these lineages are prevalent. The earliest and latest recorded dates, along with their corresponding predominant rates, are also retrieved. The obtained results are presented as an interactive Dash Table, providing an interface for applying column and row filters. This functionality allows for the exclusion of irrelevant locations or lineages based on specific research criteria. Additionally, predominant rates can be applied as a filter to exclude certain samples. Finally, based on the filtered table and the selected sequence type, all related sequences are extracted by accession number. These filtered sequences are then collected as input data for subsequent phylogeographic analysis.

Updating the sample table based on provided lineage names and sequence types, as mentioned earlier, is a crucial step in exploring input data for phylogeographic analysis. The following callback function accepts a sequence type (amino acid or nucleotide) and a list of selected lineage names as input and generates a Dash table containing relevant sample information as the output.

```
@app.callback(
    Output('lineage-table', 'data'),
    Input('button-confirm', 'n_clicks'),
    State('checklist-lineage', 'value'),
    State('dropdown-seqType', 'value')
)
def update_lineage_table(n_clicks,
                         checklist_value,
                         seqType_value):
    ...
    starts_with_conditions = " OR ".join(
            [f'n.lineage STARTS WITH "{char}"'
                 for char in checklist_value])
    query = f"""
      MATCH (n:Lineage) - [r] -> (l: Location)
      WHERE {starts_with_conditions}
      RETURN n.lineage as lineage,
             n.earliest_date as earliest_date,
             n.latest_date as latest_date,
             l.iso_code as iso_code,
             n.most_common_country as country,
             r.rate as rate
                """
    cols = ['lineage', 'earliest_date',
             'latest_date', 'iso_code',
             'country', 'rate']

    if checklist_value and seqType_value:
      #Query in Neo4j database
      # Transform Cypher results to dataframe
       df=neo_manager.queryToDataframe(query,cols)
       table_data = df.to_dict('records')
       return table_data
    ....
```

2. Location-based approach for retrieving corresponding sequences based on selected location and time period

Specific locations and a date period are defined by employing the Dash web page. Subsequently, the Neo4j database is queried to identify lineages prevalent in the specified locations during the

defined time period. The retrieved information includes the earliest and latest detected dates of the lineages in each country, along with their predominant rates. To present these findings, an interactive Dash Table is employed, facilitating the application of filters to exclude study areas or lineages below a predetermined threshold. Subsequently, the accession numbers of the corresponding sequences are extracted from the graph database. These filtered sequences are then collected for subsequent phylogeographic analysis.

The following function updates the sample table by incorporating selected start and end dates, sequence type and a list of selected locations. A Cypher query is employed to retrieve lineage data from the Neo4j database and apply filtering based on specified location and date criteria. This function empowers researchers to explore lineage data associated with diverse geographic regions within a specified date range.

```
@app.callback(
    Output('location-table', 'data'),
    Input('button-confirm', 'n_clicks'),
    State('date-range-lineage', 'start_date'),
    State('date-range-lineage', 'end_date'),
    State('checklist-location', 'value'),
    State('dropdown-seqType', 'value')
)
def update_table(n_clicks,
                 start_date,
                 end_date,
                 checklist_value,
                 seqType_value):
    ...
    query = f"""
      MATCH (n:Lineage) - [r] -> (l: Location)
      WHERE
        n.earliest_date > datetime("{start_date}")
       AND
        n.earliest_date < datetime("{end_date}")
       AND
        l.location in {checklist_value}
      RETURN n.lineage as lineage,
             n.earliest_date as earliest_date,
             n.latest_date as latest_date,
             l.iso_code,
             l.location as country,
             r.rate
                """
            cols = ['lineage', 'earliest_date',
                    'latest_date', 'iso_code',
                    'country', 'rate']
    if start_date_string and end_date_string
            and checklist_value and seqType_value:
        # Transform Cypher results dataframe
        df=neo_manager.queryToDataframe(query,cols)
        table_data = df.to_dict('records')
        return table_data
     ...
```

In summary, these approaches enable user-guided sequencing searches. Once the input sequences are defined, an Input node is generated in our graph database and appropriately labeled. This Input node establishes connections with the relevant sequencing (Nucleotide or Protein) nodes used in the analysis, highlighting relationships between the input data and the corresponding sequences. Each Input node is assigned a unique ID, which is provided for reference and traceability. These user-driven approaches provide a robust framework for sequencing searches, allowing researchers to define and explore input data relationships.

The generation of unique ID for nodes plays a crucial role in ensuring traceability for each analysis. To address this requirement, the provided function ensures that every new node is assigned a traceable ID.

```
def generate_unique_name(nodesLabel):
    driver = GraphDatabase.driver(URI,
                                  auth=("neo4j",
                                  password))
    with driver.session() as session:
        random_name = generate_short_id()

        result = session.run(
            "MATCH (u:" + nodesLabel +
            " {name: $name})
            RETURN COUNT(u)",
            name=random_name)
        count = result.single()[0]

        while count > 0:
            random_name = generate_short_id()
            result = session.run(
                "MATCH (u:" + nodesLabel +
                " {name: $name}) RETURN COUNT(u)",
                name=random_name)
            count = result.single()[0]

        return random_name
```

The following function facilitates the integration of input nodes with relationships to relevant sequence nodes within the Neo4j database, thereby enhancing the organization and management of input data and analysis entities in the network.

```
def add_Input_Neo(nodesLabel,
                  inputNode_name,
                  id_list):
    # Execute the Cypher query
    driver = GraphDatabase.driver(URI,
                                  auth=("neo4j",
                                  password))

    # Create a new node
    with driver.session() as session:
        session.run(
           "CREATE (userInput:Input {name: $name})",
           name=inputNode_name)
    # Perform MATCH query to retrieve nodes
    with driver.session() as session:
        result = session.run(
           "MATCH (n:" + nodesLabel + ")" +
           "WHERE n.accession IN $id_lt RETURN n",
           nodesLabel=nodesLabel,
           id_lt=id_list)
        # Create relationship for each matched node
        with driver.session() as session:
            for record in result:
                other_node = record["n"]
                session.run(
                   "MATCH (u:Input {name: $name}),
                          (n:" + nodesLabel +
                        " {accession: $id}) "
                    "CREATE (n)-[r:IN_INPUT]->(u)",
                    name=inputNode_name,
                    nodesLabel=nodesLabel,
                    id=other_node["accession"])
```

Parameters setting and tuning

After defining the input data, which includes sequence data and associated location information, researchers can utilize the platform to select the analysis parameters. This pivotal step entails creating an Analysis label, where the parameter values are stored as properties. These parameters encompass the step size, window size, RF distance threshold, bootstrap threshold, and the list of environmental factors involved in the analysis. Furthermore, a connection is established between the Input Node and the Analysis Node, offering several advantages. Firstly, it allows researchers to compare results obtained from the same input samples but with

different parameter settings. Secondly, it facilitates the comparison of analysis results obtained using the same parameter settings but different input samples. The interconnected Input, Analysis, and Output nodes (See Figure 1) ensure the repeatability and comparability of analysis results.

After confirming the parameters, the corresponding sequences are downloaded from NCBI [7] using the Biopython package [28], followed by performing multiple sequence alignments (MSA) [35] using the MAFFT method [36]. Subsequently, the Snakemake workflow is triggered in the backend, taking the alignment results and associated environmental data as input. Once the analysis is completed, a unique output ID is generated, enabling the results to be queried on the web platform.

The following function performs the preparation and storage of parameters and input data, subsequently triggering the workflow.

```
def trigger_workflow(df_params_geo):
    df = pd.DataFrame(df_params_geo)
    analysisNode = generate_unique_name("Analysis")
    outputNode = generate_unique_name("Output")

    # record parameters in config file
    with open('config/config.yaml', 'r') as file:
        config = yaml.safe_load(file)
    # Update the values
    config['accession_lt'] = df['id'].tolist()
    config['feature_names'] = df.columns.tolist()
    config['analysis_name'] = analysisNode
    config['output_name'] = outputNode
    # create geographic input dataset
    csv_file_name = config['geo_file']
    dff.to_csv(csv_file_name,
               index=False,
               encoding='utf-8')
    # create sequence input dataset
    aln_file_name = config['seq_file']
    seq_beforeMSA_fname = aln_file_name + '_raw'
    if config['data_type'] == 'aa':
        db_type = "protein"
    else:
        db_type = "nucleotide"
    accession_list = config['accession_lt']

    # update config dictionary to the YAML file
    with open('config/config.yaml', 'w') as file:
        yaml.dump(config, file)
    # (6) download sequences from NCBI

    seq_manager.downFromNCBI(
                   db_type,
                   accession_list,
                   seq_beforeMSA_fname)
    # (6) alignment
    seq_manager.align_MAFFT(seq_beforeMSA_fname,
                            aln_file_name)
    # (7) run aphylogeo snakemake workflow
    os.system("snakemake --cores all")
    # (8) In Neo4j create :Analysis node
    neo_manager.addAnalysisNeo()

    # (9) When Analysis finished,
     #save output dataframe into Output node
    neo_manager.addOutputNeo()
      ...
```

Output exploration

After each analysis, a unique output node is generated in the Neo4j graph database, connected to interrelated nodes that store input and parameter information, forming an intricate network of relationships. Through the ID of output node, analysis results can be conveniently traced and accessed. The platform not only facilitates querying individual results but also empowers the comparison of multiple analysis outcomes. Furthermore, as the platform is utilized, this network of input, analysis, and output nodes expands, enabling the acquisition of valuable insights from the data and facilitating comprehensive analysis of the phylogeographic patterns of SARS-CoV-2 variation.

Snakemake workflow for phylogeographic analysis

To investigate the potential correlation between the diversity of specific genes or gene fragments and their geographic distribution, a sliding window strategy was employed in addition to traditional phylogenetic analyses. As depicted in Figure 2, firstly, the multiple sequence alignment (MSA) was partitioned into windows by specifying the sliding window size and sliding window progress step size. Then a phylogenetic tree for each window was constructed. Secondly, cluster analyses for each geographic factor were performed by calculating a distance matrix and creating a reference tree based on the distance matrix and the Neighbor-Joining clustering method [37] [38]. Reference trees (based on geographic factors) and phylogenetic trees (based on sliding windows) were defined on the same set of leaves (i.e., names of species). Subsequently, the correlation between phylogenetic and reference trees was evaluated using the Robinson and Foulds (RF) distance calculation [23]. RF distances were calculated for each combination of the phylogenetic tree and the reference tree. Finally, bootstrap and RF thresholds were applied to identify gene fragments in which patterns of variation within species coincided with a particular geographic feature. These fragments can serve as informative reference points for future studies.

By scanning the complete Multiple Sequence Alignment sequences with a sliding window strategy, the phylogeographic research can effectively focus on sequence information for specific window lengths. To address the integration of genetic and environmental data, complex computational workflows are required, consisting of multiple interdependent processing steps. The aPhyloGeo snakemake workflow addresses this challenge by connecting each step through Snakemake rules, resulting in a comprehensive and easily automatable workflow. This workflow ensures reproducibility and facilitates result comparability across different sampling strategies, window sizes, and step sizes. Additionally, the aPhyloGeo workflow enables efficient processing of large datasets on parallel and distributed systems, leading to reasonable runtime.

Various tools and software were utilized to accomplish these analysis tasks, including Biopython [28], raxml-ng [39], fasttree [40], and Python libraries such as robinson-foulds, NumPy [41], and Pandas [29] [42]. A manuscript for aPhyloGeo-pipeline is available on Github Wiki (https://github.com/tahiri-lab/aPhyloGeo-pipeline/wiki).

Results and discussion

The SARS-CoV-2 virus has a genome size of approximately 30kb (See Figure 3). The first two-thirds of its genome, located at the 5'-terminal, encodes the instructions for the synthesis of two major proteins, namely pp1a, and pp1ab. Following viral enzyme processing, these proteins are transformed into 16 smaller non-structural proteins (Nsps). Specifically, ORF1a encodes nsp1–nsp10, while ORF1b encodes nsp1–nsp16, which play pivotal roles in viral replication and transcription [43]. Consequently, our first assessment of the aPhyloGeo-Covid performance focused on the pp1a region.

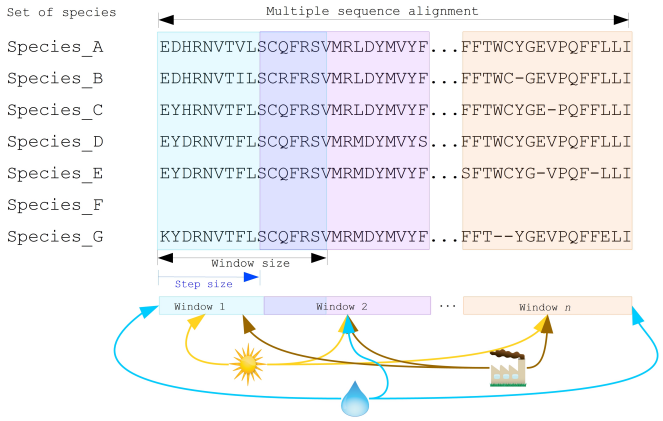

Fig. 2: *Integrated analysis of genetic data and environmental data. The aPhyloGeo workflow can analyze both amino acid sequence alignment data and nucleic acid sequence alignment data. By setting the window size and step size, the alignment of multiple sequences was segmented into sliding windows. For each sliding window, Robinson and Foulds distances are computed for every combination of the sliding window of phylogenetic tree and the reference tree created from environmental factors.*

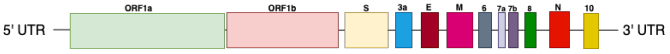

Fig. 3: *Schematic presentation of the SARS-CoV-2 genome Structure. SARS-CoV-2 follows the typical Betacoronavirus genome organization. The full-length RNA genome of approximately 29,903 nucleotides contains a replicase complex (composed of ORF1a and ORF1b) and four genes responsible for the production of structural proteins: Spike gene (S), Envelope gene (E), Membrane gene (M), and Nucleocapsid gene (N).*

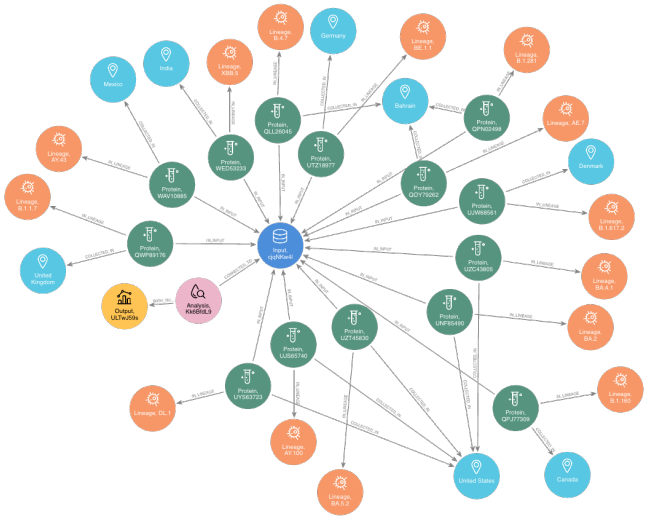

Fig. 4: *The networks of a single analysis experiment. For a specific analysis, the network highlights all entities serving as input data sources and their relationships. The Input node establishes connections between the data source objects and the specific analysis object. The Analysis node captures the parameters associated with the analysis, while the Output node stores the resulting analysis data.*

To identify and filter the appropriate datasets for further phylogeographic analysis around pp1a, 14 lineages starting with the codes AE, AY, B, BA, BE, DL, or XBB were selected from the checklist on the aPhyloGeo-Covid web page. Subsequently, with the Neo4j graph database, eight relevant locations were retrieved, where at least one of selected lineage was most prevalent (See Figure 4). An input node was created based on the lineages with connections of all the nodes of input sequences. The aPhyloGeo-Covid web page facilitated the definition of specific parameters for analysis, including a step size of 3 residues, a window size of 100 residues, an RF distance threshold of 100%, a bootstrap threshold of 0%, and a list of climate factors such as humidity, wind speed, sky shortwave irradiance, and precipitation (See Figure 5). These parameters were associated with the node of analysis and stored as properties within the node. Finally, the Snakemake workflow was triggered in the backend. At the completion of analysis, an output node with a unique identifier was generated within the Neo4j graph database (See Figure 4).

In this analysis experiment, we used aPhyloGeo-Covid to query preloaded climatic data from our Neo4j database for each sample connected to the input node. The climatic data was based on the most prevalent country and the time of initial collection. The meteorological parameters considered in our analysis included Precipitation Corrected, Relative Humidity at 2 Meters, Specific Humidity at 2 Meters, All Sky Surface Shortwave Downward Irradiance, Wind Speed within a 10-Meter Range, and Wind Speed within a 50-Meter Range. For statistical analysis, a user-defined average calculation interval of 3 days was applied. As shown in Figure 5 the 14 samples exhibited a range of precipitation from 0 mm/day to 8.57 mm/day with an average of 2.13 mm/day. The specific humidity ranged from 2.44 g/kg to 19.33 g/kg, averaging at 9.77 g/kg. The relative humidity ranged from 45.76% to 94.22%, with an average of 73.17%. Compared to other parameters, wind speed variability and sky surface shortwave downward irradiance showed relatively small variations across the samples. The sky surface shortwave downward irradiance ranged from 0.67 kW-hr/m2/day to 7.38 kW-hr/m2/day, with an average of 4.25 kW-hr/m2/day. The wind speed at 10 meters ranged from 1.90 m/s to 6.32 m/s, averaging at 3.24 m/s, while the wind speed at 50 meters ranged from 3.22 m/s to 6.40 m/s with an average of 4.39 m/s

At the end of the aPhyloGeo-Covid analysis workflow, a table was generated containing the RF distance between the phylogenetic tree of that window and the reference tree of a particular environmental feature. The distribution of normalized RF distances resulting from the phylogeographic analysis of the input dataset is presented in Figure 6. Windows exhibiting relatively lower RF distances merit further investigation. As illustrated in Figure 6, the RF distance range from 87.82% to 100%. Among the six climatic factors involved in the analysis, the sliding window region with the lower RF distance was exclusively identified in the integrated analysis involving precipitation. For this exploration, a scanning approach was employed, utilizing a window size of 100 residues and a step size of 3 residues for sequence analysis. Within the regions identified with low RF distance, special attention should be given to regions 792-940. Notably, a consistently low RF distance value of 81.82% was observed across all 17 windows spanning positions from 792 to 840. Furthermore, in accordance with SWISS-MODEL [44], the previous research validates the presence of a specific region of Nsp3 called Ubl1 (110 residues, position 819-929) within the identified sequence region. Ni et al. (2023) [45] revealed that the Ubl1 protein of SARS-CoV-2

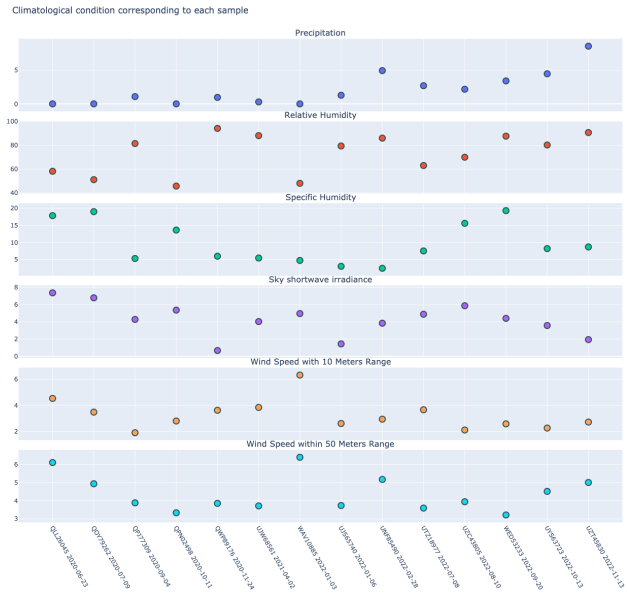

Fig. 5: *Climatic conditions of each sample in most common country at the time of first collection. The climate factors involved include Precipitation Corrected (mm/day), Relative Humidity at 2 Meters (%), Specific Humidity at 2 Meters (g/kg), All Sky Surface Shortwave Downward Irradiance (kW-hr/m^2/day), Wind Speed within 10 Meters Range (m/s), Wind Speed within 50 Meters Range (m/s).*

exhibits competitive binding with RNA molecules to the N protein, resulting in the dissociation of viral ribonucleoprotein complexes. Based on these findings, they propose a model that explains how the N protein binding to the Ubl1 domain of Nsp3 leads to the dissociation of viral ribonucleoprotein complexes.

Our phylogeography-based exploration revealed a notable correlation between mutations in the region [792-940] and precipitation. As a reproducible phylogeographic platform, aPhyloGeo-Covid offers the potential to expand the sample size for further investigation and facilitates the comparability of analysis results.

In addition of correlation analysis between correlation diversity of subfregment of gene and climate condition, we also inferred the ORF1a phylogeny and window regions 792-940 of ORF1a using the RAxML-NG method [39], and then conducted a detailed horizontal gene transfer (HGT) and recombination analyses (See Figure 7) using the HGT-Detection program available on the T-Rex web server [46]. The HGT-Detection program allows one to infer all possible horizontal gene transfer events for a given group of species by reconciling the species tree (i.e. ORF1a gene tree in our case) with different gene phylogenies built for regions of individual genes [47] [48]. Significantly, every identified horizontal gene transfer event can be understood from three perspectives: Firstly, it may signify a distinct complete or partial HGT occurrence between genetically distant species. Secondly, it could indicate the occurrence of parallel evolution, where the involved species underwent similar genetic changes independently. Lastly, it could also indicate the emergence of a new species (referred to as a gene transfer recipient) resulting from the recombination of the donor species genome with that of a neighboring recipient in the species' evolutionary history [49].

The minimum-cost transfer scenario with five HGTs necessary to reconcile the variants and gene phylogenies is shown in Figure 7 (HGTs are depicted by numbered arrows). The analysis initially

TABLE 2: *Putative horizontal gene transfer events in the window regions of 792-940 residue (amino acid sequences) of 14 SARS-Cov-2 variants. Each iteration of the horizontal gene transfer (HGT) algorithm is accompanied by the Robinson-Foulds distance (RF) and bipartition distance (BD) values were calculated. The HGT transfer occurs from the origin of the subtree to the destination of the subtree.*

Iteration	RF distance	BD distance	Origin Subtree	Destination Subtree
1	10	7.5	QWP89176	WAV10885
2	6	3.5	QLL26045	(QPJ77309, QWP89176, WAV10885)
3	4	2.5	UJS65740	(QLL26045, QPJ77309, QPN02498, QWP89176, UJW68561, WAV10885)
4	2	1.5	UTZ18977	UNF85490
5	0	0.0	(UNF85490, UTZ18977)	UZC43805

measured the Robinson and Foulds distance (RF) between the phylogenetic tree of ORF1a and the inferred phylogenetic trees of the window regions 792-940 of ORF1a, yielding a dissimilarity of 16. Five iterations led to the identification of HGT events (See Table 2): the first iteration detected an HGT from subtree QWP89176 to subtree WAV10885 (RF = 10 and BD = 7.5), followed by an HGT from subtree QLL26045 to subtrees QPJ77309, QWP89176, and WAV10885 (RF = 6 and BD = 3.5). The third iteration revealed an HGT from subtree UJS65740 to subtrees QLL26045, QPJ77309, QPN02498, QWP89176, UJW68561, and WAV10885 (RF = 4 and BD = 2.5). In the fourth iteration, an HGT event occurred from subtree UTZ18977 to subtree UNF85490 (RF = 2 and BD = 1.0). Finally, the fifth iteration showed an HGT from subtrees UNF85490 and UTZ18977 to subtree UZC43805 (RF = 0 and BD = 0.0). Overall, five HGT events were identified throughout the analysis.

Conclusions and future work

This project demonstrates the creation of an open-source, interactive platform that aims to enhance phylogeographic research. By integrating graph databases and a modularized Snakemake workflow, the platform effectively addresses the challenges posed by manual tools, streamlining the extraction, validation, and integration of genetic and environmental data. The platform primarily focuses on advancing the analysis of geographic and environmental data associated with SARS-CoV-2.

The utilization of the platform leads to the accumulation of diverse findings, contributed by researchers conducting various analyses. As more researchers join the platform, this network of data sources and analysis outputs continues to expand. The centralized database acts as a repository, providing researchers with access to a wide range of results and facilitating exploration and knowledge sharing within the scientific community. Although the platform is currently undergoing testing, it is expected that the interconnectedness of analyses will increase as the platform gains popularity and attracts more researchers. This network enables researchers to compare their findings and identify meaningful patterns. Overall, the platform facilitates the dissemination of research findings, encourages collaboration and building upon each

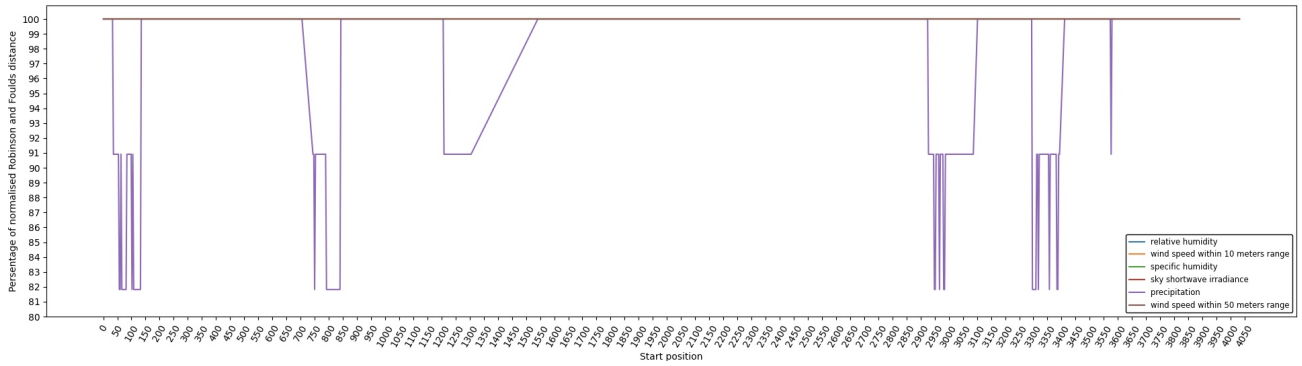

Fig. 6: *Variation of normalized Robinson and Foulds (RF) distance on the Multiple Sequence Alignment (MSA) for different climate facters. A sliding window approach with a window size of 100 residues and a step size of 3 residues was applied. X-axis indicates the start position of sliding windows on the MSA. Various colors represent six analysed climate factors which are relative humidity (blue), specific humidity (green), wind speed within 10 meters range (yellow), wind speed within 50 meters range (brown), sky shortwave irradiance (red), and precipitation (purple).*

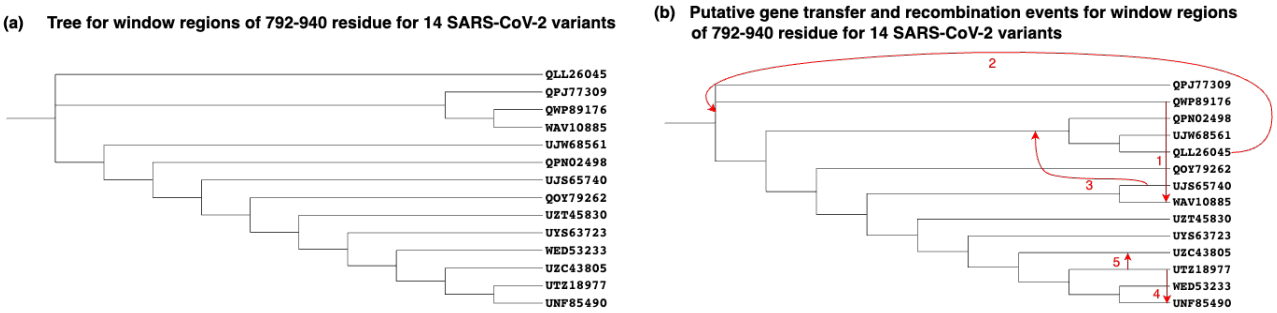

Fig. 7: *Putative horizontal gene transfer events found for the window regions of 792-940 residue (amino acid sequences) of 14 SARS-Cov-2 variants. (a) presents the phylogenetic tree of the window regions 792-940 of ORF1a. (b) presents the phylogenetic tree of ORF1a (amino acid sequences) with putative horizontal gene transfers mapped into it.*

previous work, and fosters a sense of community and scientific advancement.

To further enhance aPhyloGeo-Covid, several potential avenues for improvement can be explored:

1) Expanding the scope of available data resources, with a specific focus on augmenting geographic and environmental data. By enriching and diversifying the dataset, the aPhyloGeo-Covid project can unlock greater potential to uncover valuable insights regarding the dynamics of SARS-CoV-2 transmission and its intricate relationship with geographical and environmental variables.
2) Broadening the scope of phylogeographic analysis and comprehensively investigating the evolutionary dynamics and spatial spread of the virus can be achieved by expanding the existing pipeline of aPhyloGeo-Covid. In addition to the current pipeline, which focuses on exploring the correlation between specific genes or gene fragments and their geographic distribution, incorporating additional phylogeographic analysis workflows is recommended. By incorporating a diverse range of analysis approaches, aPhyloGeo-Covid can offer a more extensive toolkit for studying the evolutionary dynamics and spatial dissemination of SARS-CoV-2. This expanded toolkit will contribute to a more comprehensive understanding of the virus and its transmission patterns.
3) To meet the increasing research demands and accommodate larger datasets, prioritizing scalability and efficiency is crucial in the development of aPhyloGeo-Covid. Enhancing scalability and efficiency will enable the platform to handle substantial volumes of data while maintaining optimal performance. This capability is vital for researchers and public health practitioners, as it ensures fast and reliable analyses, even as the data continues to grow. By ensuring scalability and efficiency, aPhyloGeo-Covid can effectively support decision-making processes and provide valuable insights into the spatial spread and evolution of SARS-CoV-2.

Acknowledgements

The authors thank SciPy conference and reviewers for their valuable comments on this paper. This work was supported by the Natural Sciences and Engineering Research Council of Canada, the Université de Sherbrooke grant, and the Centre de recherche en écologie de l'Université de Sherbrooke (CREUS).

References

[1] S. Dellicour, C. Troupin, F. Jahanbakhsh, A. Salama, S. Massoudi, M. K. Moghaddam, G. Baele, P. Lemey, A. Gholami, and H. Bourhy, "Using phylogeographic approaches to analyse the dispersal history, velocity and direction of viral lineages—application to rabies virus spread in iran," *Molecular ecology*, vol. 28, no. 18, pp. 4335–4350, 2019, https://doi.org/10.1111/mec.15222.

[2] C. B. Vogels, D. E. Brackney, A. P. Dupuis, R. M. Robich, J. R. Fauver, A. F. Brito, S. C. Williams, J. F. Anderson, C. B. Lubelczyk, R. E. Lange *et al.*, "Phylogeographic reconstruction of the emergence and spread of powassan virus in the northeastern united states," *Proceedings of the National Academy of Sciences*, vol. 120, no. 16, p. e2218012120, 2023, https://doi.org/10.1073/pnas.2218012120.

[3] G. Franzo, G. Faustini, M. Legnardi, M. Cecchinato, M. Drigo, and C. M. Tucciarone, "Phylodynamic and phylogeographic reconstruction of porcine reproductive and respiratory syndrome virus (prrsv) in europe: Patterns and determinants," *Transboundary and Emerging Diseases*, vol. 69, no. 5, pp. e2175–e2184, 2022, https://doi.org/10.1111/tbed.14556.

[4] A. Munsey, F. N. Mwiine, S. Ochwo, L. Velazquez-Salinas, Z. Ahmed, F. Maree, L. L. Rodriguez, E. Rieder, A. Perez, S. Dellicour *et al.*, "Phylogeographic analysis of foot-and-mouth disease virus serotype o dispersal and associated drivers in east africa," *Molecular ecology*, vol. 30, no. 15, pp. 3815–3825, 2021, https://doi.org/10.1111/mec.15991.

[5] E. C. Holmes, "The phylogeography of human viruses," *Molecular ecology*, vol. 13, no. 4, pp. 745–756, 2004, https://doi.org/10.1046/j.1365-294X.2003.02051.x.

[6] Á. O'Toole, V. Hill, O. G. Pybus, A. Watts, I. I. Bogoch, K. Khan, J. P. Messina, T. COVID, B.-U. C. G. Network, H. Tegally *et al.*, "Tracking the international spread of sars-cov-2 lineages b. 1.1. 7 and b. 1.351/501y-v2 with grinch," *Wellcome Open Research*, vol. 6, 2021, https://doi.org/10.12688/wellcomeopenres.16661.2.

[7] J. R. Brister, D. Ako-Adjei, Y. Bao, and O. Blinkova, "Ncbi viral genomes resource," *Nucleic acids research*, vol. 43, no. D1, pp. D571–D577, 2015, https://doi.org/10.1093/nar/gku1207.

[8] S. Khare, C. Gurry, L. Freitas, M. B. Schultz, G. Bach, A. Diallo, N. Akite, J. Ho, R. T. Lee, W. Yeo *et al.*, "Gisaid's role in pandemic response," *China CDC weekly*, vol. 3, no. 49, p. 1049, 2021, https://doi.org/10.46234/ccdcw2021.255.

[9] O. A. Marzouk, "Assessment of global warming in al buraimi, sultanate of oman based on statistical analysis of nasa power data over 39 years, and testing the reliability of nasa power against meteorological measurements," *Heliyon*, vol. 7, no. 3, p. e06625, 2021, https://doi.org/10.1016/j.heliyon.2021.e06625.

[10] E. Mathieu, H. Ritchie, E. Ortiz-Ospina, M. Roser, J. Hasell, C. Appel, C. Giattino, and L. Rodés-Guirao, "A global database of covid-19 vaccinations," *Nature human behaviour*, vol. 5, no. 7, pp. 947–953, 2021, https://doi.org/10.1038/s41562-021-01122-8.

[11] J. E. McCormack, S. M. Hird, A. J. Zellmer, B. C. Carstens, and R. T. Brumfield, "Applications of next-generation sequencing to phylogeography and phylogenetics," *Molecular phylogenetics and evolution*, vol. 66, no. 2, pp. 526–538, 2013, https://doi.org/10.1016/j.ympev.2011.12.007.

[12] A. McGaughran, L. Liggins, K. A. Marske, M. N. Dawson, L. M. Schiebelhut, S. D. Lavery, L. L. Knowles, C. Moritz, and C. Riginos, "Comparative phylogeography in the genomic age: Opportunities and challenges," *Journal of Biogeography*, vol. 49, no. 12, pp. 2130–2144, 2022, https://doi.org/10.1111/jbi.14481.

[13] J. Guia, V. G. Soares, and J. Bernardino, "Graph databases: Neo4j analysis." in *ICEIS (1)*, 2017, pp. 351–356, https://doi.org/10.5220/0006356003510356.

[14] S. Timón-Reina, M. Rincón, and R. Martínez-Tomás, "An overview of graph databases and their applications in the biomedical domain," *Database*, vol. 2021, 2021, https://doi.org/10.1093/database/baab026.

[15] W. Laddada, C. Zanni-Merk, and L. F. Soualmia, "Analyzing sars-cov-2 sequence patterns by semantic trajectories." *Stud Health Technol Inform*, pp. 197–200, 2022, https://doi.org/10.3233/SHTI220696.

[16] R. Angles, "A comparison of current graph database models," in *2012 IEEE 28th International Conference on Data Engineering Workshops*. IEEE, 2012, pp. 171–177, https://doi.org/10.1109/ICDEW.2012.31.

[17] C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y. Chen, and D. Wilkins, "A comparison of a graph database and a relational database: a data provenance perspective," in *Proceedings of the 48th annual Southeast regional conference*, 2010, pp. 1–6, https://doi.org/10.1145/1900008.1900067.

[18] O. Uphyrkina, W. E. Johnson, H. Quigley, D. Miquelle, L. Marker, M. Bush, and S. J. O'Brien, "Phylogenetics, genome diversity and origin of modern leopard, panthera pardus," *Molecular ecology*, vol. 10, no. 11, pp. 2617–2633, 2001, https://doi.org/10.1046/j.0962-1083.2001.01350.x.

[19] S.-J. Luo, J.-H. Kim, W. E. Johnson, J. v. d. Walt, J. Martenson, N. Yuhki, D. G. Miquelle, O. Uphyrkina, J. M. Goodrich, H. B. Quigley *et al.*, "Phylogeography and genetic ancestry of tigers (panthera tigris)," *PLoS biology*, vol. 2, no. 12, p. e442, 2004, https://doi.org/10.1371/journal.pbio.0020442.

[20] D. J. Taylor, S. J. Connelly, and A. A. Kotov, "The intercontinental phylogeography of neustonic daphniids," *Scientific Reports*, vol. 10, no. 1, p. 1818, 2020, https://doi.org/10.1038/s41598-020-58743-8.

[21] M. A. Aziz, O. Smith, H. A. Jackson, S. Tollington, S. Darlow, A. Barlow, M. A. Islam, and J. Groombridge, "Phylogeography of panthera tigris in the mangrove forest of the sundarbans," *Endangered Species Research*, vol. 48, pp. 87–97, 2022, https://doi.org/10.3354/esr01188.

[22] A. Koshkarov, W. Li, M.-L. Luu, and N. Tahiri, "Phylogeography: Analysis of genetic and climatic data of sars-cov-2," 2022, https://doi.org/10.25080/majora-212e5952-018.

[23] D. F. Robinson and L. R. Foulds, "Comparison of phylogenetic trees," *Mathematical biosciences*, vol. 53, no. 1-2, pp. 131–147, 1981, https://doi.org/10.1016/0025-5564(81)90043-2.

[24] J. Köster and S. Rahmann, "Snakemake—a scalable bioinformatics workflow engine," *Bioinformatics*, vol. 28, no. 19, pp. 2520–2522, 2012, https://doi.org/10.1093/bioinformatics/bty350.

[25] V. Jalili, E. Afgan, Q. Gu, D. Clements, D. Blankenberg, J. Goecks, J. Taylor, and A. Nekrutenko, "The galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2020 update," *Nucleic acids research*, vol. 48, no. W1, pp. W395–W402, 2020, https://doi.org/10.1093/nar/gkaa434.

[26] V. Spišaková, L. Hejtmánek, and J. Hynšt, "Nextflow in bioinformatics: Executors performance comparison using genomics data," *Future Generation Computer Systems*, 2023, https://doi.org/10.1016/j.future.2023.01.009.

[27] L. Wratten, A. Wilm, and J. Göke, "Reproducible, scalable, and shareable analysis pipelines with bioinformatics workflow managers," *Nature methods*, vol. 18, no. 10, pp. 1161–1168, 2021, https://doi.org/10.1038/s41592-021-01254-9.

[28] P. J. Cock, T. Antao, J. T. Chang, B. A. Chapman, C. J. Cox, A. Dalke, I. Friedberg, T. Hamelryck, F. Kauff, B. Wilczynski *et al.*, "Biopython: freely available python tools for computational molecular biology and bioinformatics," *Bioinformatics*, vol. 25, no. 11, pp. 1422–1423, 2009, https://doi.org/10.1093/bioinformatics/btp163.

[29] Wes McKinney, "Data Structures for Statistical Computing in Python," in *Proceedings of the 9th Python in Science Conference*, Stéfan van der Walt and Jarrod Millman, Eds., 2010, pp. 56 – 61, https://doi.org/10.25080/Majora-92bf1922-00a.

[30] P. Lemenkova, "Processing oceanographic data by python libraries numpy, scipy and pandas," *Aquatic Research*, vol. 2, no. 2, pp. 73–91, 2019, https://doi.org/10.3153/AR19009.

[31] E. L. Hatcher, S. A. Zhdanov, Y. Bao, O. Blinkova, E. P. Nawrocki, Y. Ostapchuck, A. A. Schäffer, and J. R. Brister, "Virus variation resource–improved response to emergent viral outbreaks," *Nucleic acids research*, vol. 45, no. D1, pp. D482–D490, 2017, https://doi.org/10.1093/nar/gkw1065.

[32] S. Hossain, C. Calloway, D. Lippa, D. Niederhut, and D. Shupe, "Visualization of bioinformatics data with dash bio," in *Proceedings of the 18th Python in Science Conference*, vol. 126. SciPy, Austin, Texas, pp. 126–133, 2019, p. 133, https://doi.org/10.25080/Majora-7ddc1dd1-012.

[33] V. Liermann and S. Li, "Dynamic dashboards," in *The Digital Journey of Banking and Insurance, Volume II: Digitalization and Machine Learning*. Springer, 2021, pp. 155–180, https://doi.org/10.1007/978-3-030-78829-2_9.

[34] G. Jordan and G. Jordan, "Neo4j+ python," *Practical Neo4j*, pp. 169–213, 2014, https://doi.org/10.1007/978-1-4842-0022-3_9.

[35] R. C. Edgar and S. Batzoglou, "Multiple sequence alignment," *Current opinion in structural biology*, vol. 16, no. 3, pp. 368–373, 2006, https://doi.org/10.1016/j.sbi.2006.04.004.

[36] K. Katoh and D. M. Standley, "Mafft multiple sequence alignment software version 7: improvements in performance and usability," *Molecular biology and evolution*, vol. 30, no. 4, pp. 772–780, 2013, https://doi.org/10.1093/molbev/mst010.

[37] N. Saitou and M. Nei, "The neighbor-joining method: a new method for reconstructing phylogenetic trees." *Molecular biology and evolution*, vol. 4, no. 4, pp. 406–425, 1987, https://doi.org/10.1093/oxfordjournals.molbev.a040454.

[38] R. Mihaescu, D. Levy, and L. Pachter, "Why neighbor-joining works," *Algorithmica*, vol. 54, pp. 1–24, 2009, https://doi.org/10.1007/s00453-007-9116-4.

[39] A. M. Kozlov, D. Darriba, T. Flouri, B. Morel, and A. Stamatakis, "Raxml-ng: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference," *Bioinformatics*, vol. 35, no. 21, pp. 4453–4455, 2019, https://doi.org/10.1093/bioinformatics/btz305.

[40] M. N. Price, P. S. Dehal, and A. P. Arkin, "Fasttree: computing large minimum evolution trees with profiles instead of a distance matrix," *Molecular biology and evolution*, vol. 26, no. 7, pp. 1641–1650, 2009, https://doi.org/10.1093/molbev/msp077.

[41] S. Van Der Walt, S. C. Colbert, and G. Varoquaux, "The numpy array: a structure for efficient numerical computation," *Computing in science & engineering*, vol. 13, no. 2, pp. 22–30, 2011, https://doi.org/10.1109/MCSE.2011.37.

[42] J. Bernard and J. Bernard, "Python data analysis with pandas," *Python Recipes Handbook: A Problem-Solution Approach*, pp. 37–48, 2016, https://doi.org/10.1007/978-1-4842-0241-8_5.

[43] M. T. Khan, M. Irfan, H. Ahsan, A. Ahmed, A. C. Kaushik, A. S. Khan, S. Chinnasamy, A. Ali, and D.-Q. Wei, "Structures of sars-cov-2 rna-binding proteins and therapeutic targets," *Intervirology*, vol. 64, no. 2, pp. 55–68, 2021, https://doi.org/10.1159/000513686.

[44] A. Waterhouse, M. Bertoni, S. Bienert, G. Studer, G. Tauriello, R. Gumienny, F. T. Heer, T. A. P. de Beer, C. Rempfer, L. Bordoli *et al.*, "Swiss-model: homology modelling of protein structures and complexes," *Nucleic acids research*, vol. 46, no. W1, pp. W296–W303, 2018, https://doi.org/10.1093/nar/gky427.

[45] X. Ni, Y. Han, R. Zhou, Y. Zhou, and J. Lei, "Structural insights into ribonucleoprotein dissociation by nucleocapsid protein interacting with non-structural protein 3 in sars-cov-2," *Communications Biology*, vol. 6, no. 1, p. 193, 2023, https://doi.org/10.1038/s42003-023-04570-2.

[46] A. Boc, A. B. Diallo, and V. Makarenkov, "T-rex: a web server for inferring, validating and visualizing phylogenetic trees and networks," *Nucleic acids research*, vol. 40, no. W1, pp. W573–W579, 2012, https://doi.org/10.1093/nar/gks485.

[47] A. Boc and V. Makarenkov, "Towards an accurate identification of mosaic genes and partial horizontal gene transfers," *Nucleic acids research*, vol. 39, no. 21, pp. e144–e144, 2011, https://doi.org/10.1093/nar/gkr735.

[48] E. Denamur, G. Lecointre, P. Darlu, O. Tenaillon, C. Acquaviva, C. Sayada, I. Sunjevaric, R. Rothstein, J. Elion, F. Taddei *et al.*, "Evolutionary implications of the frequent horizontal transfer of mismatch repair genes," *Cell*, vol. 103, no. 5, pp. 711–721, 2000, https://doi.org/10.1016/S0092-8674(00)00175-6.

[49] V. Makarenkov, B. Mazoure, G. Rabusseau, and P. Legendre, "Horizontal gene transfer and recombination analysis of sars-cov-2 genes helps discover its close relatives and shed light on its origin," *BMC ecology and evolution*, vol. 21, pp. 1–18, 2021, https://doi.org/10.1186/s12862-020-01732-2.