Geographic Name Resolution Service: A tool for the standardization and indexing of world political division names, with applications to species distribution modeling

Brad L. Boyle[1], Brian S. Maitner[2], George G. C. Barbosa[1], Rohith K. Sajja[1], Xiao Feng[3], Cory Merow[2], Erica A. Newman[4], Daniel S. Park[5,6], Patrick R. Roehrdanz[7], Brian J. Enquist[1,8]

[1]Department of Ecology and Evolutionary Biology, University of Arizona, Tucson, AZ 85721, USA

[2]Eversource Energy Center and Department of Ecology and Evolutionary Biology, University of Connecticut, Storrs, CT 06269, USA

[3]Department of Geography, Florida State University, Tallahassee, FL 32306, USA

[4]School of Natural Resources & the Environment, University of Arizona, Tucson, AZ 85721, USA

[5]Department of Biological Sciences, Purdue University, West Lafayette, IN 47907, USA

[6]Purdue Center for Plant Biology, Purdue University, West Lafayette, IN 47907, USA

[7]The Moore Center for Science, Conservation International, Arlington, VA 22202, USA

[9]The Santa Fe Institute, USA, 1399 Hyde Park Rd, Santa Fe, NM 87501, USA

* Corresponding author

E-mail: bboyle@arizona.edu

Abstract

Massive biological databases of species occurrences, or georeferenced locations where a species has been observed, are essential inputs for modeling present and future species distributions. Location accuracy is often assessed by determining whether the observation geocoordinates fall within the boundaries of the declared political divisions. This otherwise simple validation is complicated by the difficulty of matching political division names to the correct geospatial object. Spelling errors, abbreviations, alternative codes, and synonyms in multiple languages present daunting name disambiguation challenges. The inability to resolve political division names reduces usable data and analysis of erroneous observations can lead to flawed results.

Here, we present the Geographic Name Resolution Service (GNRS), an application for the correction, standardization and indexing of world political division names. The GNRS resolves political division names against a reference database that combines names and codes from GeoNames with geospatial object identifiers from the Global Administrative Areas Database (GADM). In a trial resolution of political division names extracted from >270 million species occurrences, only 1.9%, representing just 6% of occurrences, matched exactly to GADM political divisions in their original form. The GNRS was able to resolve, completely or in part, 92% of the remaining 378,568 political division names, or 86% of the full biodiversity occurrence dataset. In an assessment of geocoordinate accuracy for >239 million species occurrences, resolution of political divisions by the GNRS enabled detection of an order of magnitude more errors and an order of magnitude more error-free occurrences. By providing a novel solution to a major data quality impediment, the GNRS liberates a tremendous amount of biodiversity data for quantitative biodiversity research. The GNRS runs as a web service and can be accessed via an API, an R package, and a web-based graphical user interface. Its modular architecture is easily integrated into existing data validation workflows.

Introduction

Large databases of georeferenced species occurrences (GSOs) are fueling an increasingly diverse body of research into past, current and future patterns of species distributions and traits [1]. GSOs provide essential inputs for species distribution models (SDMs) [2–5], which in turn have been used to predict relative vulnerability of species and populations to climate change [6], identify priority conservation strategies [7] and assess the biodiversity impacts of policies governing land use, deforestation and burning [8]. SDMs and the raw GSOs from which they are derived are helping to clarify distributions of disease vector organisms and identify new disease hotspots [9,10]. GSOs and associated trait data from museum specimens have been used to disentangle patterns of temporal and spatial change in body size of birds [11] and melanism in butterflies [12]. Given the breadth of applications of SDMs, it is crucial that they are robust, which in turn depends on the accuracy of the species occurrence data that drive them. The challenge is how to best identify, differentiate, and potentially correct erroneous or inaccurate geographic distribution information.

The fitness of GSOs for such analyses hinges on the accuracy of the associated location data. Despite recent advances in automated tools for standardization and correction of errors, the potential presence of erroneous or inaccurate geo coordinates in biodiversity “big data" remains a major concern [13,14]. A widely used method for assessing reliability of coordinates is to check if they fall within the boundaries of their associated political divisions (hereafter, “geopolitical validation”). A point falling outside a declared political division is flagged for inspection and either corrected or rejected [15]. Another common validation links a GSO via its declared political division to one or more country, state or county taxonomic checklists to determine if the species is native or introduced in the region of observation; observations of introduced species may be excluded from further analysis [16], unless modeling of invasive species distributions is the focus of the research [17]. A surprising impediment to these otherwise simple validations is lack of standardization among political division names, identifiers and hierarchies.

The importance of political divisions as both units of data aggregation and data quality pitfalls extends well beyond GSOs and SDMs. Analysis of relationships between environmental factors, health care policy and health care outcomes are a mainstay of public health research, with many studies relying on data aggregated by first- and second-level administrative divisions [19]. Multi-country comparisons of crime statistics aggregated at subnational levels are common in criminology and sociological research [20]. A recent study of human reliance on protected natural areas throughout the global tropics combined geospatial information on protected areas with household survey data aggregated by subnational administrative units [18]. Incomplete or inconsistent standardization of political division names and identifiers increases the burden of data aggregation, especially when historical data are involved [21,22].

A promising way forward is the development of a general tool for the standardization of political division names, identifiers and hierarchies. However, this goal is complicated by the myriad of alternative names, spellings and abbreviations used to refer to the same country or subnational unit. In addition, geographical data processing codes, such as ISO (International Organization for Standardization; [23], FIPS (the United States' Federal Information Processing Standards; [24], and HASC (Hierarchical Administrative Subdivision Codes; [25] may be used instead of names. Multiple languages, accented characters, and different character set encodings provide additional layers of complexity. Spelling errors may also be introduced during data entry. Together, these issues represent a daunting disambiguation challenge on par with taxonomic name resolution [26]. Failure to resolve political division names can lead to loss of data by exclusion of GSOs of unknown quality or the inability to georeference historical observations [27]. Naive use of unvalidated GSOs can result in misleading, erroneous or biased research results [28].

Here, we describe a software tool for the correction, standardization and indexing of world political division names, the Geographic Name Resolution Service (GNRS). The GNRS accepts one or more 1-3 level political division name combinations (country, country+state or country+state+county, or equivalent), and resolves them against the Database of Global Administrative Areas (GADM; [29] and GeoNames [30], supplemented with additional names and codes from Natural Earth [31]. For each name resolved, the application returns the standard GADM name, a plain-ascii English-language name (minus class identifiers such "State of", "Provincia de", "Département", and so on), ISO codes, and Geonames and GADM identifiers. (The GNRS remains neutral with respect to the validity of political division names and competing jurisdictional claims). Match scores and summaries describing how the submitted name was matched and overall matching completeness are also returned with the resolved name. Other GNRS options support retrieval of alternative names in multiple languages and character sets from both Geonames and GADM. Standardized political division names and GADM identifiers can be used to retrieve spatial objects from GADM to perform political geovalidation of GSOs, or to submit the GSO to other validation services such as the BIEN Native Species Resolver [16]. Despite the importance of reference databases of administrative district spatial objects (such as GADM) and names (such as GeoNames), standardizing and indexing data against applying these databases remains challenging due to lack of standardization of object names and identifiers and incomplete linkages among reference databases. To our knowledge, no existing service links these databases and provides the informatics tools for resolving large volumes of unstandardized data against them. Our goal in developing the GNRS is to fill this gap.

Overview of the GNRS

Architecture

Originally developed as part of the BIEN database pipeline [16,26,32,33], the GNRS is the first of a series of data validation and standardization tools which we are making available to the biodiversity research community as modular web services. Each service will be accessible through a variety of interfaces, using standardized plain text input and output that allows multiple services to be chained together into more complex validations. We are releasing these services as standalone applications to enable independent development of each service and to encourage scrutiny and improvement of algorithms and data by the community.

All BIEN validations services share the common architecture shown in Fig 1. Components of the architecture include: (1) a core service in which user data are standardized against a reference database using algorithms implemented primarily in SQL; (2) a data integration application which builds and periodically updates the reference database (a partly normalized "data warehouse" *sensu Inmon [34]* from external sources, (3) a controller layer which manages concurrent requests and implements parallelization using makeflow [35]; (4) an administrative interface which interacts directly with the controller; (5) a JSON-based application programming interface (API) which supports large input-output data payloads; (6) an R package; and (7) a web-based graphical user interface. All public access to the core service, including via the R package and web interface, is handled by the API. The GNRS runs in the Linux environment and was developed under Ubuntu 16.04.7 LTS [36].

Several design elements of the BIEN validation service architecture optimize processing of very large data sets within a multi-user environment. These performance features are described in Supporting Information (S1 Appendix 1).

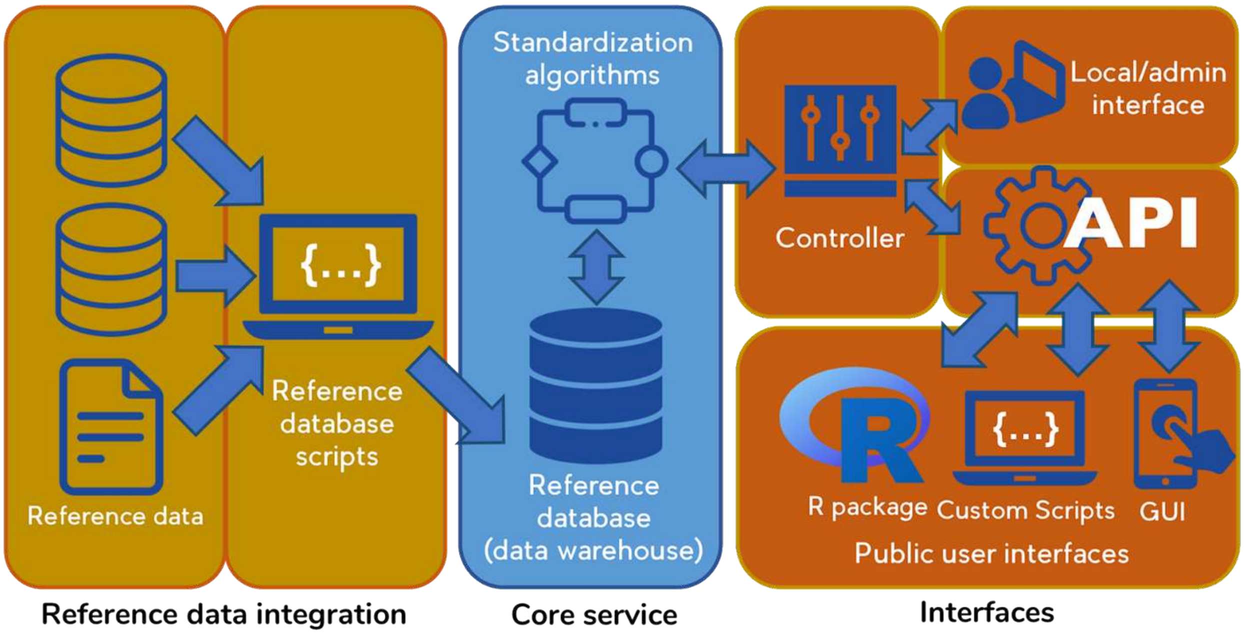

Fig 1. BIEN validation service architecture, as implemented for the GNRS. Reference data are stored locally within the core service as a periodically-updated, versioned data warehouse. A controller manages parallelization and optimization of concurrent requests. Interfaces include a JSON API, an R package, and a web-based graphical user interface. All public interaction with the core service is handled by the API.

Reference database

Political division names are resolved by the GNRS against a reference database consisting of the names, codes and identifiers of all countries plus their first-level (state/province) and second-level (county/parish) administrative divisions in GADM. In addition, each GADM political division is linked to a lookup table of alternative names in multiple languages from Geonames, supplemented with additional codes from the Natural Earth and a custom list of name variants prepared by the authors (the latter included in the GNRS source code repository; see https://github.com/ojalaquellueva/gnrs/tree/master/gnrs_db/data). Names, codes and identifiers for country-, state- and county-level political divisions from these sources are merged within a single PostgreSQL database [37] by a pipeline of SQL statements managed by Bash shell commands [38]. The steps and challenges involved in merging these data sources are described in detail in Supporting Information (S1 Appendix 2).

Metadata

Management of metadata within the GNRS database and transmission via user interfaces follows the principles established by the BIEN database and its public interface, the BIEN R package [16]. Summary tables within the GNRS reference database manage information on reference data sources and the GNRS itself. Versions, date of access, source URLs, project websites and bibtex-formatted citations for GADM, GeoNames and NaturalEarth are stored in table “source”. Metadata on the GNRS (database release date, code version and citation) is maintained in table “meta”. Metadata on other contributors of resources or data is stored in table “collaborator”. All metadata can be queried via the API using routes “source”, “meta” and “collaborator”; this information is also exposed by the RGNRS R package and displayed on the GNRS website.

User input

The basic input for the GNRS is a 1- to 3-level political division combination (PDC) consisting of a country, a 1st-level administrative division (state, province, department, etc.) and a 2nd-level administrative division (county, parish, municipality, etc.), separated by commas. Country is required but 1st- and 2nd-level divisions are optional; however, a 1st-level division must be present if a 2nd-level division is supplied. Both comma delimiters must be present, even if one or more administrative division is absent. Names which themselves contain commas must be surrounded by double quotes. Each PDC must be on its own line. Examples of data suitable for input to the GNRS are shown in Table 1.

Table 1. Input format for political divisions submitted to the GNRS via the web user interface (https://gnrs.biendata.org/) and API. Format requirements for the GNRS R package are the same as the API (see documentation).

Interface	Examples
Web	USA,Arizona,Pima County
	México,Oaxaca,
	Costa Rica,,
	Guyana,Upper Takutu-Upper Essequibo,"Yakarinta - Wowetta, Surama"
API (with id)	1,USA,Arizona,Pima County
	2,México,Oaxaca,
	3,Costa Rica,,
	4,Guyana,Upper Takutu-Upper Essequibo,"Yakarinta - Wowetta, Surama"
API (no id)	,USA,Arizona,Pima County
	,México,Oaxaca,
	,Costa Rica,,
	,Guyana,Upper Takutu-Upper Essequibo,"Yakarinta - Wowetta, Surama"

One or more PDCs in this format can be submitted directly to the GNRS web interface by pasting them into the input box (Fig 2). The input format for the GNRS API and R package is similar to the basic format, with the exception of an additional, user-supplied unique identifier ("user_id") in the first column (i.e., user_id, country, state_province, county_parish). A single-column identifier provides a reliable way of joining the multi-column GNRS output back to databases, where NULL values in some fields can result in failed joins and potential loss of data. Use of identifiers is optional; however, the four column format must be maintained in the order described, with all three comma delimiters present on all lines, as shown in the API examples in Table 1. Data are submitted to the API as JSON attached to the body of a POST request. The R package automatically handles the conversion to JSON and construction of the API request.

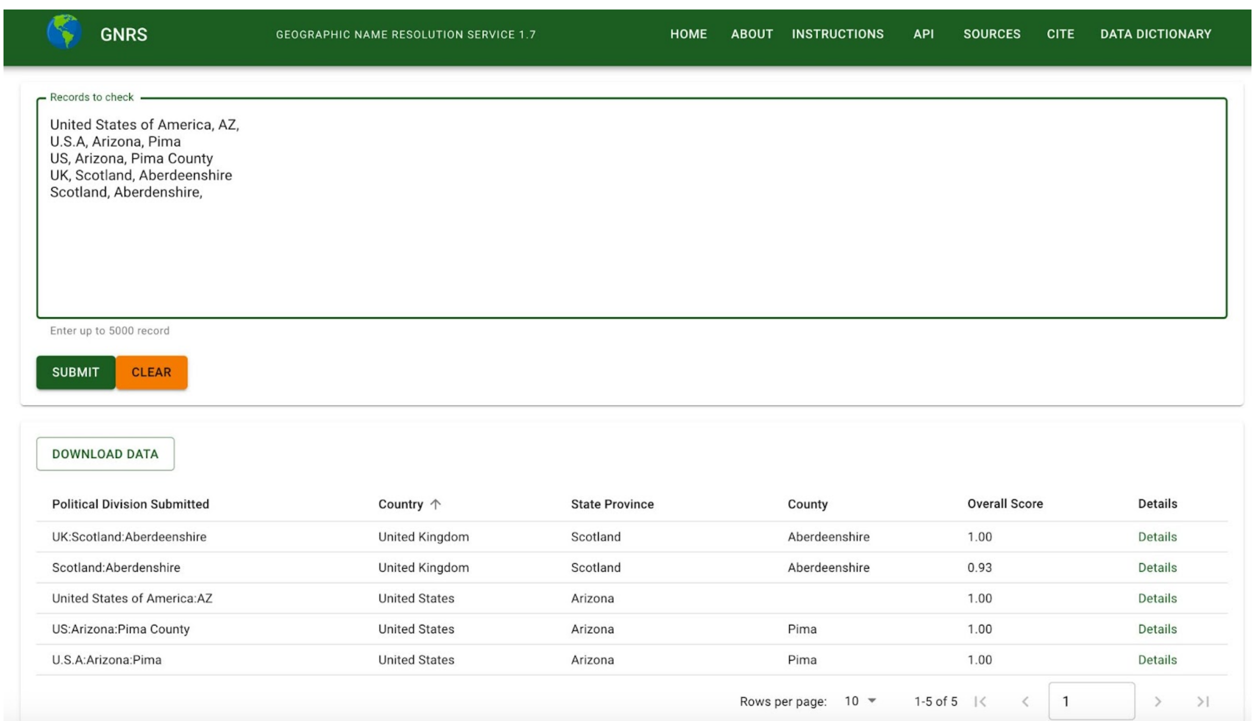

Fig 2. Screenshot of the GNRS web user interface. Comma delimited political divisions are pasted into the top input box. Paginated results, displayed below the input, can be sorted by any column and downloaded as comma- or tab-delimited files. The "Details" hyperlink at the end of each results row displays the full results for that row, along with field definitions from the GNRS data dictionary.

Name resolution workflow

Political divisions are resolved by working down the 3-level political division hierarchy beginning with country. The algorithm first tries matching by code (ISO, FIPS, HASC) before attempting to match unresolved political divisions by standard and alternative names. After all exact matching methods have been exhausted, fuzzy matching of the remaining unresolved names is attempted using the Postgres implementation of trigram similarity [39]. The GNRS uses a default trigram match threshold of 0.5, a conservative setting which we have found favors avoidance of false positives. The match threshold can also be adjusted on the fly by the users of the R package or API. At each step, the match method used for a successful match is saved and returned to the user in fields "match_method_country", "match_method_state_province" and "match_method_county_parish" (example values: "iso code", "exact match standard name", "fuzzy match alternate name"). A match score from 0 to 1 (where 0="no match" and 1="exact match") is calculated as the trigram similarity between the name submitted and the name matched and saved to fields "match_score_country", "match_score_state_province" and "match_score_county_parish". After all matching steps have been completed, an "overall_score" is calculated as the average of the match scores of all submitted political divisions. One of three descriptors of overall match completeness ("no match", "partial match", "full match") is saved to field "match_status". The GNRS also returns the political division level submitted and the political division level matched, using terms "country", "state_province" and "county_parish".

Two classes of political divisions not resolved by the default GNRS workflow required custom solutions. These are (1) territories and other subnational geopolitical units treated as countries ("states-as-countries") by the reference databases, and (2) countries belonging to multinational unions, with the latter treated as countries and its member countries treated as first-level divisions ("countries-as-states"). An example of state-as-country is Puerto Rico, an unincorporated territory of the United States which is treated by GADM and GeoNames as a top-level political entity with ISO code PR, but frequently recorded in biodiversity data as a state-level division of the United States (e.g., "USA, Puerto Rico"). Examples of countries-as-states are England, Scotland and other member countries of the United Kingdom, which are treated as 1st-level political divisions of the UK by GADM and GeoNames, but which also appear as countries in biodiversity data. The GNRS solutions to these special cases are described in Supporting Information (S1 Appendix 3).

Interfaces

Linux command line

Developers installing their own instance of the GNRS can invoke the GNRS directly from the Linux shell using commands `gnrs_batch.sh` (single batch mode) or `gnrs_par.pl` (parallel mode). See the main README in the GNRS GitHub repository for documentation of syntax and usage examples (https://github.com/ojalaquellueva/gnrs/blob/master/README.md). Accessing the GNRS directly via the shell bypasses the API and its default limit of 5000 rows per request, thus enabling processing of very large data sets in a single operation.

GNRS API

All public interaction with the GNRS—including requests from the GNRS R package and GNRS website—is handled by a JSON-based API (see [40] with a single route (https://gnrs.biendata.org/gnrs_api.php). Different endpoints and their parameters are specified in JSON object "opts" (options). Request data are converted to JSON object "data". Objects opts and data are combined into a single nested JSON object and attached to the body of the POST request submitted to the API.

API endpoint "resolve" performs name resolution of the political divisions contained in the POST data; it supports a single optional parameter "tfuzzy" which accepts a numeric value between 0 and 1 and allows the user to vary the default trigram fuzzy match score threshold. Other API endpoints include a data dictionary defining all name resolution output fields; detailed lists of names, alternate names, codes and identifiers of countries, states and counties; and metadata and citations for the GNRS and its sources. Descriptions of all API endpoints are provided in Table 2. Example scripts demonstrating calls to the GNRS API in R and PHP are provided in the api subdirectory of the GitHub GNRS repository (https://github.com/ojalaquellueva/gnrs/tree/master/api). Results are returned to the user as JSON. The GNRS API is written in PHP [41].

Table 2. GNRS API endpoints and their meanings. Endpoints, parameters and input data are attached to the body of the POST request as a nested JSON object, with the endpoint and its parameters in element "option" and input data in element "data".

Endpoint	**Purpose**	**Data**
resolve	Resolve submitted political division and return standardized names and identifiers	One or more sets of political divisions (country plus up to 2 lower political divisions) optionally preceded by user-supplied record identifiers
countrylist	Return names and identifiers of all countries in GNRS	none
statelist	Return names and identifiers of all states in submitted countries.	List of countries (name) for which to return states. Get country names from route "countrylist"
countylist	Return names and identifiers of all counties in submitted states.	List of states (GNRS identifier state_province_id) for which to return counties. Values of state_province_id from route "statelist"
meta	Return metadata on the GNRS	none
sources	Return metadata on reference data sources used by the GNRS	none
citations	Return citations for the GNRS and all data sources	none

Endpoint	Purpose	Data
dd	Return definitions of output fields (data dictionary)	none

GNRS R package

The R package GNRS provides a family of functions for interacting with the GNRS API using the R language [42]. All major functionality available by calling the GNRS API directly is also available through the R package (Table 3). GNRS can be installed from CRAN using the command `install.packages("GNRS")` or the development version can be installed directly from the GNRS GitHub repository (https://github.com/EnquistLab/RGNRS) using the devtools package [43] with the command `devtools::install_github("EnquistLab/RGNRS")`. The GNRS package relies on the package httr [44] to interact with the API, jsonlite [45] to convert to and from json, and the packages knitr, rmarkdown, devtools, and testthat [46–49] for development and testing. GNRS R package functions begin with the prefix "`GNRS_…`" to simplify function location through tab-completion.

Table 3. GNRS R package functionality

API Option	R function	Input data	Purpose
resolve	`GNRS()`	Political division dataframe containing 4 columns: user_id, country ,state_province, and county_parish. Number of batches (Optional)	Resolve submitted political division and return standardized names and identifiers
resolve	`GNRS_super_simple()`	country, state_province (Optional), county_parish (Optional), user_id (Optional)	Resolve submitted political division and return standardized names and identifiers
countrylist	`GNRS_get_countries()`	none	Return names and identifiers of all countries in GNRS
statelist	`GNRS_get_states()`	country_id (Optional)	Return names and identifiers of all states, or states in submitted countries (if country_id is supplied).

API Option	R function	Input data	Purpose
countylist	`GNRS_get_counties()`	state_province_id (Optional)	Return names and identifiers of all counties, or counties in submitted states (if state_province_id is supplied).
meta	`GNRS_version()`	none	Return version metadata on the GNRS
sources	`GNRS_sources()`	none	Return metadata on reference data sources used by the GNRS
citations	`GNRS_citations()`	none	Return citations for the GNRS and all data sources
dd	`GNRS_data_dictionary()`	none	Return definitions of output fields (data dictionary)
	`GNRS_metadata()`	bibtex_file (Optional)	Wrapper function that returns metadata on version, sources, acknowledgements, and citations.

API Option	R function	Input data	Purpose
	`GNRS_template()`	nrow (Optional)	Returns an empty dataframe of nrow (default is 1) rows that can be populated by users.

GNRS web interface

The GNRS website is a graphical user interface to the GNRS that runs on both desktop and mobile devices (Fig 2). Political divisions are pasted or typed directly into an input box and results are displayed below. Results may be downloaded in comma-delimited (CSV) or tab-delimited (TSV) formats. With the exception of user IDs, which are not supported, most functionality available via the GNRS API and GNRS R package is also available through the GNRS website. Metadata served via API options “meta”, “sources”, “citations” and “dd” (data dictionary) are displayed on pages “Cite”, “Sources” and “Data dictionary”. The GNRS website was developed using the open-source Next.js framework, written in JavaScript with the back-end using [50] runtime and the front-end using the React library [51]. The interface was designed using Material-UI, an open-source React-component library that follows Material Design principles [52].

Documentation

The main README in the GNRS repository (https://github.com/ojalaquellueva/gnrs/blob/master/README.md) documents GNRS installation and configuration, command-line syntax for invoking the GNRS core service from the Linux shell, format requirements for input data, and definitions of all output returned by the GNRS. Working examples using sample data included in the repository are also provided. Example files in the API subdirectory of the GNRS GitHub repository (https://github.com/ojalaquellueva/gnrs/tree/master/api) demonstrate how to interact with the GNRS API in PHP (https://github.com/ojalaquellueva/gnrs/blob/master/api/gnrs_api_example.php) and in R without using the GNRS R package (https://github.com/ojalaquellueva/gnrs/blob/master/api/gnrs_api_example.R). The GNRS website includes a short tutorial on how to use the web interface (https://gnrs.biendata.org/instructions/). Examples demonstrating usage of the GNRS R package are provided below.

Sample workflow with the GNRS R package

Example 1: A few political divisions

GNRS_super_simple() is the quickest method of standardizing a small number of political division names. This function does not require that the user supply a dataframe, and instead takes character vectors as input.

```
library("GNRS")
GNRS_super_simple("USA")
GNRS_super_simple(country = c("USA", "Canada"))
```

```
GNRS_super_simple(country = "USA",
                  state_province = "AZ",
                  county_parish = "Pima County")
```

Example 2: Many political divisions

In most cases, users will have existing data sets containing political division names that they wish to standardize. In this case, the user only has to generate an appropriately formatted dataframe from their dataset. This can be done manually, or the function `GNRS_template()` can be used to generate an empty dataframe that can then be populated. Here, we demonstrate this using the data that come packaged with the GNRS R package (accessed through the `data()` function).

```
data("gnrs_testfile")
gnrs_dataframe <- GNRS_template(nrow = nrow(gnrs_testfile))
gnrs_dataframe$country <- gnrs_testfile$country
gnrs_dataframe$state_province <- gnrs_testfile$state_province
gnrs_dataframe$county_parish <- gnrs_testfile$county_parish
clean_dataframe <- GNRS(political_division_dataframe = gnrs_dataframe)
metadata <- GNRS_metadata()
```

In both examples, the function `GNRS_metadata()` is usually the last step and is used to extract information that is needed for publication (e.g. version number, citation information). Complete input-output for this and the preceding example is provided in Supporting Information (S1 Appendix 4).

Case study: Validating species occurrences from the BIEN database

This example illustrates the challenges of working with political divisions from biodiversity data from multiple sources and the ability of prior name resolution by the GNRS to improve the effectiveness of downstream validations such as geopolitical validation.

Methods

As part of the validation workflow for the BIEN 4.2 database we extracted all distinct, verbatim country-, state- and county-level political divisions (declared PDCs; see "User input" for definition) from the >270 million species occurrence records in the BIEN biodiversity observations database [16,32]. The BIEN 4.2 observations consist of herbarium specimen data and vegetation plot occurrence records assembled from 4,946 distinct data sources ranging from individual research data sets to herbarium collections databases to large aggregators of regional and global biodiversity data (see [33] for details). The occurrence records included both georeferenced and non-georeferenced observations. We then assessed performance of the GNRS by comparing the numbers of declared political division names matching exactly to political divisions in GADM in their original form to those matching after resolution by the GNRS.

To explore the consequences of political division name resolution for downstream validation of geocoordinate accuracy, we used the subset of BIEN 4.2 species observations accompanied by geocoordinates (BIEN GSOs) to compare rates of mismatch between the declared political divisions and the political divisions indicated by the accompanying geocoordinates ("observed political divisions"). For each GSO, we determined the observed country, state and county by joining its coordinates to spatial object representations of world administrative divisions in the

GADM database. Spatial joins and retrieval of GADM political division identifiers were performed using the BIEN GVS (see https://github.com/ojalaquellueva/gvs). GADM identifiers (gid_0, gid_1 and gid_2) of the observed political divisions were then compared to the GADM identifiers of the declared political divisions. A GSO with all observed political divisions matching all corresponding declared political divisions was classified as having passed validation; a GSO with one or more sets of non-matching observed and declared political division identifiers was classified as having failed. This validation is equivalent to testing if the GSO's coordinates fall within its declared political divisions. We performed validation twice: once using the GADM identifiers retrieved by an exact match of the verbatim declared political division name to names stored natively in the GADM database, and a second time using the GADM identifiers retrieved by resolution of declared political division names using the GNRS.

Results & Discussion

Political division name resolution

A total of 409,797 unique verbatim PDCs were extracted from the 271,188,222 species observations in the BIEN 4.2 database. After processing by the GNRS, 163,174 (39.8%) of the unique PDCs were fully matched, 234,452 (57.2%) were partly matched and 12,171 (3.0%) returned no match, where "fully matched" PDCs had all declared political division names matching exactly to GADM political division names and "partly matched" PDCs had one or more unmatched political division names. Of the fully matched PDCs, only 7,593 (1.9% of total PDCs), representing 16,138,042 (6.0%) of total observations matched exactly as submitted (that is, all verbatim political division names matched exactly to names in the GADM database). The remaining 155,581 fully matched PDCs (38.0% of total PDCs) required resolution by the GNRS—either by exact matching on codes, exact matching on alternative names and spellings, or fuzzy matching on standard and alternative names—to recover the corresponding GADM administrative units. Of the partly matched names, 222,987 also required some degree of resolution by the GNRS. Thus the GNRS was able to resolve, in part or completely, 378,568 initially non-matching PDCs (92.4% of the total PDCs) representing 232,270,686 observations, or 85.6% of the total biodiversity observation data set.

After resolution by the GNRS, 30% of country names (429), 65.6% state names (78,710) and 58.6% (175,312) of county names remained unresolved. However, unmatched country and state names represented only >0.1% (263,300) and 6.2% (16,870,530) of total observations, respectively. Unmatched county-level names accounted for the majority of observations with partly or completely unresolved PDCs (72,273,914, or 26.7% of total observations). At all levels, most unmatched names appeared to be unresolvable genuine errors such as informal regions (e.g., “Europe”, “Indochina”), locality descriptions (especially in the state field) and other information unrelated to political division names (Supporting Information, Tables S5-S8). However, many unmatched county-level names contained valid, correctly-spelled names preceded or followed by administrative level 2 type identifiers (e.g., “Oblast”, “Prefecture”, “District”, etc.) or their abbreviations (“Obl.”, “Pref.”, “Distr.”, “Cty.”, etc.). Although the GNRS attempts to remove such class identifiers prior to matching, the reference tables used to detect type identifiers are incomplete; matching of county-level names could be increased by expanding these tables, in particular by adding commonly-used abbreviations (see S1 Appendix 5 for details).

Geocoordinate validation with and without the GNRS

A total of 239,662,948 species observations had non-null values of latitude and longitude on range [0:90] and [-180:180], respectively, allowing validation of the declared political divisions against the observed political divisions determined by their coordinates. The global distributions of points passing and failing validation, with and without prior resolution of declared political division names by the GNRS, are shown in Fig 3.

Of the GSOs passing validation, 19,123,498 (8.0% of total georeferenced GSOs) had declared political divisions correct as submitted (Fig 3B) compared to 220,762,855 (92.1%) passing validation after name resolution by the GNRS (Fig 3D). Of the GSOs failing validation, 1,122,010 (0.5%) had declared political divisions correct as submitted (Fig 3A), compared to the 14,771,429 (6.2%) erroneous GSOs detected after name resolution by the GNRS (Fig 3C). Thus, prior name resolution by the GNRS enabled detection of an additional 201,639,357 correct observations and exclusion of an additional 13,649,419 erroneous observations.

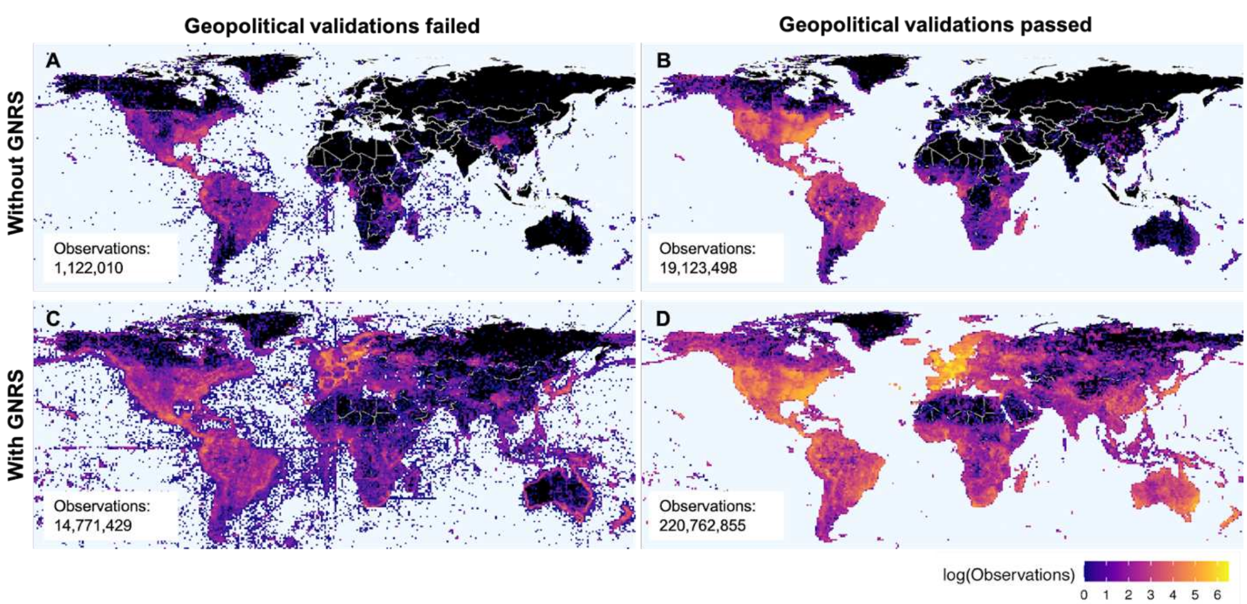

Fig 3. Geopolitical validation results for 240 million georeferenced species occurrences, with and without prior political division name resolution by the GNRS. Using the GNRS allows you to detect and reject an order of magnitude more bad points (A vs. C) and detect and accept an order of magnitude more valid points (B vs. D). Colors represent density of georeferenced observations per 1 x 1 degree cells failing or passing validation that observation coordinates fall within the declared political divisions. (A) Political divisions correct as submitted,

validation failed; (B) political divisions correct as submitted, validation passed; (C) political divisions correct or resolved by GNRS, validation failed; (D) political divisions correct or resolved by GNRS, validation passed. Black=zero observations.

The increase in data validated following resolution by the GNRS had strong spatial components, with especially striking increases in Africa, Asia and Australia. For some regions, such as central Australia, density of validated GSOs increased from near zero to hundreds of thousands following political division name resolution by the GNRS. Overall, name resolution by the GNRS prior to geopolitical validation increased the number of observations validated (passed and failed) by an order of magnitude, from 20,245,508 to 235,534,284.

Caveats

An important caveat to bear in mind when using the GNRS is that its data sources encompass modern countries only. For example, Yugoslavia is not present in GADM or GeoNames; historical collections from “Yugoslavia” will therefore not be resolved by the GNRS and will not be available for downstream validation. In addition, historical changes to extant country boundaries are not represented. For example, collections from the region of South Sudan collected prior to 2011 (the year of South Sudan’s independence from Sudan) would most likely bear the country name “Sudan”. Although the GNRS will resolve the latter name, subsequent validation of the associated coordinates using GADM would locate the point of observation in modern “South Sudan”, resulting in rejection of the occurrence as invalid. Future development of the GNRS may address data validation of GSOs derived from historical collections. This challenge will require reference data that includes historical geopolitical entities and their start and end dates [53] in addition to modern political entities. Users would need to submit dates of observation in addition to geocoordinates and declared political divisions.

A second caveat is that the GNRS currently resolves GADM spatial object identifiers only. However, it also returns a variety of standard political division codes such as ISO 3166, FIPS and HASC, which can in turn be used to retrieve spatial object identifiers from other widely used administrative division databases such as Natural Earth [31]. Future releases of the GNRS will store spatial object identifiers for additional sources natively within the GNRS database and expose this information to users.

Conclusions

Political division name resolution is a critical but often neglected step in verifying the accuracy of species occurrence data. Political geovalidation is of limited value if political divisions are misspelled or represented by codes and spelling variants not present in the geospatial reference data. The GNRS fills this gap by rapidly standardizing political division names, synonyms and codes against widely-used administrative division reference data sets. A variety of interfaces enable use of the GNRS by users with different skill levels and programming abilities (including non-programmers) and simplifies integration into existing data quality pipelines. As demonstrated by a case study involving >239 million georeference species occurrences, prior name resolution by the GNRS can enable validation of an order of magnitude more error-free data and detection of an order of magnitude more erroneous data, compared to using unresolved political division names and codes.

Software and data availability

A publicly available instance of the GNRS API can be accessed directly at https://gnrsapi.xyz/gnrs_api.php, or indirectly using the GNRS R package or GNRS web interface. The GNRS R package can be downloaded from GitHub (https://github.com/EnquistLab/RGNRS) using the devtools package [48], from CRAN (see https://cran.r-project.org/web/packages/GNRS/index.html). The GNRS web interface can be accessed at https://gnrs.biendata.org.

Source code for the GNRS database, core services and API are available from the GNRS GitHub repository (https://github.com/ojalaquellueva/gnrs). Example scripts demonstrating how to call the API in R and PHP are available from the API subdirectory of the GNRS repository (https://github.com/ojalaquellueva/gnrs/tree/master/api). Code to import GADM content to PostgreSQL is available from https://github.com/ojalaquellueva/gadm. Code to import GeoNames is available from https://github.com/ojalaquellueva/geonames. Source code for the GNRS web interface is available from https://github.com/EnquistLab/GNRSweb. All source code for all GNRS components is freely available under MIT licenses.

Access to the complete GNRS database is governed by the licenses of the contributing databases, which range from CC0 (NaturalEarth) to CC BY (GeoNames) to the equivalent of CC BY-NC-ND (GADM). Limitations imposed by the GADM license prohibits us from directly redistributing the complete copies of the GNRS database. However, users can build an identical copy of the GNRS database using the GNRS source code with data obtained directly from the contributing databases. Step-by-step instructions are provided in the GNRS GitHub repository.

An example data file for testing the GNRS is available here: https://github.com/ojalaquellueva/gnrs/blob/master/data/user/gnrs_testfile.csv.

Data and code for replicating the GNRS case study (Fig 3 and all summary statistics) are available from doi:10.5281/zenodo.6370837.

Supporting Information

S1 File. Additional GNRS application details, analyses and examples.

Acknowledgements

We gratefully acknowledge the authors and administrators of GADM, GeoNames and Natural Earth for compiling, maintaining and distributing the data resources that made this project possible. GNRS parallelization code was originally developed by Naim Matasci for the Taxonomic Name Resolution Service [26]. The staff at NCEAS and CyVerse provided critical computational support.

Author contributions

Conceptualization: BLB, BJE. Software – GNRS database, search engine and API: BLB. Software – GNRS R package: BSM, BLB. Software – GNRS web interface: GGCB, RKS, BLB. Writing – Original Draft Preparation: BLB. Writing – Figures: BLB, EAN. Writing – Review & Editing: All authors.

References

1. Antonelli A, Ariza M, Albert J, Andermann T, Azevedo J, Bacon C, et al. Conceptual and empirical advances in Neotropical biodiversity research. PeerJ. 2018 Oct 4;6:e5644.

2. Guisan A, Zimmermann NE. Predictive habitat distribution models in ecology. Ecol Modell. 2000;135(2-3):147–86.

3. Guisan A, Thuiller W. Predicting species distribution: offering more than simple habitat models. Ecol Lett. 2005 Sep;8(9):993–1009.

4. Peterson AT, Soberón J, Pearson RG, Anderson RP, Martínez-Meyer E, Nakamura M, et al. Ecological Niches and Geographic Distributions (MPB-49). Princeton University Press; 2011. 328 p.

5. Franklin J. Mapping Species Distributions: Spatial Inference and Prediction. Cambridge University Press; 2010. 339 p.

6. Willis SG, Foden W, Baker DJ, Belle E, Burgess ND, Carr JA, et al. Integrating climate change vulnerability assessments from species distribution models and trait-based approaches. Biol Conserv. 2015 Oct 1;190:167–78.

7. Hannah L, Roehrdanz PR, Marquet PA, Enquist BJ, Midgley G, Foden W, et al. 30% Land Conservation and Climate Action Reduces Tropical Extinction Risk By More Than 50%. Ecography . 2020;1–11.

8. Feng X, Merow C, Liu Z, Park DS, Roehrdanz PR, Maitner B, et al. How deregulation, drought and increasing fire impact Amazonian biodiversity. Nature [Internet]. 2021 Sep 1; Available from: http://dx.doi.org/10.1038/s41586-021-03876-7

9. Foley DH, Weitzman AL, Miller SE, Faran ME, Rueda LM, Wilkerson RC. The value of georeferenced collection records for predicting patterns of mosquito species richness and endemism in the Neotropics. Ecol Entomol. 2007 Nov 27;0(0):071203162814003 – ???

10. Carlson CJ, Albery GF, Merow C, Trisos CH, Zipfel CM, Eskew EA, et al. Climate change will drive novel cross-species viral transmission [Internet]. Available from: http://dx.doi.org/10.1101/2020.01.24.918755

11. Weeks BC, Willard DE, Zimova M, Ellis AA, Witynski ML, Hennen M, et al. Shared morphological consequences of global warming in North American migratory birds. Ecol Lett. 2020 Feb;23(2):316–25.

12. MacLean HJ, Nielsen ME, Kingsolver JG, Buckley LB. Using museum specimens to track morphological shifts through climate change. Philos Trans R Soc Lond B Biol Sci [Internet]. 2018 Nov 19;374(1763). Available from: http://dx.doi.org/10.1098/rstb.2017.0404

13. Serra-Diaz JM, Enquist BJ, Maitner B, Merow C, Svenning J-C. Big data of tree species distributions: how big and how good? Forest Ecosystems. 2018 Jan 15;4(1):30.

14. Park DS, Davis CC. Implications and alternatives of assigning climate data to geographical centroids. J Biogeogr. 2017 Oct 28;44(10):2188–98.

15. Zizka A, Silvestro D, Andermann T, Azevedo J, Duarte Ritter C, Edler D, et al. CoordinateCleaner: Standardized cleaning of occurrence records from biological collection databases. Methods Ecol Evol. 2019;10(5):744–51.

16. Maitner BS, Boyle B, Casler N, Condit R, Donoghue J, Durán SM, et al. The bien r package: A tool to access the Botanical Information and Ecology Network (BIEN) database. Methods Ecol Evol. 2017;2017(July):1–7.

17. Barbet-Massin M, Rome Q, Villemant C, Courchamp F. Can species distribution models really predict the expansion of invasive species? PLoS One. 2018 Mar 6;13(3):e0193085.

18. Fedele G, Donatti CI, Bornacelly I, Hole DG. Nature-dependent people: Mapping human direct use of nature for basic needs across the tropics. Glob Environ Change. 2021 Nov 1;71:102368.

19. Wang F. Why Public Health Needs GIS: A Methodological Overview. Ann GIS. 2020;26(1):1–12.

20. Piatkowska SJ, Hövermann A. A Culture of Hostility and Crime Motivated by Bias: A Cross-National Multilevel Analysis of Structural Influences. International Criminal Justice Review. 2019 Jun 1;29(2):141–67.

21. Foa RS. Decentralization, historical state capacity and public goods provision in Post-Soviet Russia. World Dev. 2022 Apr 1;152:105807.

22. Faye CM, Wehrmeister FC, Melesse DY, Mutua MKK, Maïga A, Taylor CM, et al. Large and persistent subnational inequalities in reproductive, maternal, newborn and child health intervention coverage in sub-Saharan Africa. BMJ Glob Health. 2020 Jan 26;5(1):e002232.

23. ISO 3166 [Internet]. 2021 [cited 2021 Sep 8]. Available from: https://www.iso.org/iso-3166-country-codes.html

24. Federal Information Processing Standards Publications (FIPS PUBS) [Internet]. 2021 [cited Accessed: Sep 08 2021]. Available from: https://www.nist.gov/itl/publications-0/federal-information-processing-standards-fips

25. Law G. Administrative Subdivisions of Countries: A Comprehensive World Reference, 1900 through 1998. McFarland; 2010. 463 p.

26. Boyle B, Hopkins N, Lu Z, Raygoza Garay JA, Mozzherin D, Rees T, et al. The taxonomic name resolution service: an online tool for automated standardization of plant names. BMC Bioinformatics. 2013 Jan;14(1):16.

27. Burgio KR, Carlson CJ, Tingley MW. Lazarus ecology: Recovering the distribution and migratory patterns of the extinct Carolina parakeet. Ecol Evol. 2017 Jul;7(14):5467–75.

28. Qian H. Are species lists derived from modeled species range maps appropriate for macroecological studies? A case study on data from BIEN. Basic Appl Ecol. 2020 Nov 1;48:146–56.

29. University of California, Berkeley, Museum of Vertebrate Zoology. Global Administrative Areas (GADM) [Internet]. GADM maps and data. 2018 [cited 2018 May 5]. Available from: http://www.gadm.org

30. Geonames. GeoNames [Internet]. 2020 [cited 2020 Apr 20]. Available from: https://www.geonames.org/

31. Kelse NV, Patterson T, Furno D, Buckingham T, Springer N, Cross L. Natural Earth [Internet]. Natural Earth. 2020 [cited 2020 Apr 15]. Available from: https://www.naturalearthdata.com

32. Enquist BJ, Condit R, Peet B, Schildhauer M, Thiers B, Bien WG. Cyberinfrastructure for an integrated botanical information network to investigate the ecological impacts of global climate change on plant biodiversity. PeerJ Preprints. 2016;No. e2615:1–32.

33. Enquist BJ, Feng X, Boyle B, Maitner B, Newman EA, Jørgensen PM, et al. The commonness of rarity: Global and future distribution of rarity across land plants. Sci Adv. 2019 Nov;5(11):eaaz0414.

34. Inmon WH. Building the data warehouse. John wiley & sons; 2005.

35. Albrecht M, Donnelly P, Bui P, Thain D. Makeflow: A portable abstraction for cluster, cloud, and grid computing. Technical Report TR-2011--02 [Internet]. 2011; Available from: http://www.cse.nd.edu/Reports/2011/TR-2011-02.pdf

36. Enterprise Open Source and Linux [Internet]. [cited 2021 Sep 9]. Available from: https://ubuntu.com/

37. PostgreSQL Global Development Group. PostgreSQL [Internet]. 2021 [cited 2021 Sep 9]. Available from: https://www.postgresql.org/

38. Bash - GNU Project - Free Software Foundation [Internet]. 2021 [cited 2021 Sep 9]. Available from: https://www.gnu.org/software/bash/

39. Angell RC, Freund GE, Willett P. Automatic spelling correction using a trigram similarity measure. Inf Process Manag. 1983 Jan 1;19(4):255–61.

40. JSON:API Latest Specification (v1.0) [Internet]. 2021 [cited 2021 Sep 23]. Available from: https://jsonapi.org/format/

41. PHP: Hypertext Preprocessor [Internet]. 2021 [cited 2021 Sep 17]. Available from: https://www.php.net/

42. Venables WN, Smith DM, Team RDC, Others. An introduction to R [Internet]. Citeseer; 2009. Available from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.462.8971&rep=rep1&type=pdf

43. Wickham H, Chang W. Devtools: Tools to make developing r packages easier. R package version. 2016;1(0):9000.

44. Wickham H. Tools for Working with URLs and HTTP [R package httr version 1.4.2]. 2020 Jul 20 [cited 2021 Sep 9]; Available from: https://CRAN.R-project.org/package=httr

45. Ooms J. The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects [Internet]. arXiv:1403. 2805 [stat. CO]. 2014. Available from: https://arxiv.org/abs/1403.2805

46. Xie Y. knitr: a comprehensive tool for reproducible research in R. In: Implementing reproducible research. Chapman and Hall/CRC; 2018. p. 3–31.

47. Allaire JJ, Xie Y, McPherson J, Luraschi J, Ushey K, Atkins A, et al. rmarkdown: Dynamic Documents for R [Internet]. 2020. Available from: https://github.com/rstudio/rmarkdown

48. Wickham H, Hester J, Chang W. devtools: Tools to Make Developing R Packages Easier [Internet]. 2020. Available from: https://CRAN.R-project.org/package=devtools

49. Wickham H. testthat: Get Started with Testing [Internet]. Vol. 3, The R Journal. 2011. p. 5–10. Available from: https://journal.r-project.org/archive/2011-1/RJournal_2011-1_Wickham.pdf

50. Node.js [Internet]. [cited 2021 Sep 9]. Available from: https://nodejs.org/en/

51. React [Internet]. [cited 2021 Sep 9]. Available from: https://reactjs.org/

52. Material Design [Internet]. 2021 [cited 2021 Oct 27]. Available from: https://material.io/

53. Weidmann NB, Kuse D, Gleditsch KS. The Geography of the International System: The CShapes Dataset. International Interactions. 2010 Feb 26;36(1):86–106.