

dataFishing: An efficient Python tool and user-friendly web-form for mining mitochondrial and chloroplast sequences, taxonomic, and biodiversity data

Luan Rabelo^{a,b,*}, Davidson Sodré^c, Oscar David Albito Balcázar^a, Murilo Furtado do Rosário^a, Aurycéia Jaquelyne Guimarães-Costa^{a,d}, Grazielle Gomes^e, Iracilda Sampaio^a, Marcelo Vallinoto^{a,f,*}

^a Laboratório de Evolução (LEVO), Instituto de Estudos Costeiros (IECOS), Universidade Federal do Pará (UFPA), Campus de Bragança, Bragança, Pará, Brazil

^b Instituto Tecnológico Vale (ITV), Belém, Pará, Brazil

^c Universidade Federal Rural da Amazônia (UFRA), Campus de Capitão Poço, Capitão Poço, Brazil

^d AFYA, Faculdade de Ciências Médicas, Bragança, Pará, Brazil

^e Laboratório de Genética Aplicada (LAGA), Universidade Federal do Pará, Campus de Bragança, IECOS, Bragança, Brazil

^f CIBIO-InBIO, Centro de Investigação em Biodiversidade and Recursos Genéticos, Universidade do Porto, Porto, Portugal

ARTICLE INFO

Keywords:

Taxonomy
Genetic data
Multirepository
Bioinformatics

ABSTRACT

NCBI GenBank and BOLD Systems are important databases for biodiversity research, in which the deposited data can be used for various purposes, such as species identification analysis, evolutionary studies, biodiversity monitoring, as well as assessing the effects of possible climate changes on species distributions. Other information, such as taxonomy, collection site locations, and conservation status, is often critical for these studies. Some databases, such as GBIF, BOLD Systems, and GenBank, provide data on the taxonomy, habitat, and geographic distribution of various taxonomic groups, while others, such as WoRMS and IUCN, have specific data on marine species and conservation status. However, depending on the taxonomic group studied, searches in these databases can encompass dozens or hundreds of queries, forcing researchers to conduct extensive searches in each database, which is a time-consuming and error-prone process. To facilitate and automate access to this information, we introduce dataFishing, a Python script and a web form. dataFishing is faster and more efficient than other R packages, such as *bold*, *taxize*, *rgbif*, *rredlist*, and *worms*, for obtaining taxonomic information from the consulted databases. Moreover, it allows the retrieval of DNA sequences, common names, synonyms, conservation status, and species occurrence points. This tool is free and will enable a more systematized and time-efficient search, which tends to facilitate such data inquiries.

1. Introduction

Mitochondrial and chloroplast sequence information housed in databases is a fundamental resource for studies on global biodiversity (Nakazato, 2020). These databases allow for the retrieval of important genetic information ranging from sequences of a single gene (Bucklin et al., 2021; Ito et al., 2022; Vallinoto et al., 2004) to a combination of nuclear and mitochondrial genes (Dias et al., 2018; Louro et al., 2021; Rocha et al., 2015; Sequeira et al., 2020) or even complete genomes (Muhala et al., 2024a; Song et al., 2022; Xu et al., 2023). This information enables various analyses, such as species identification (da Costa et al., 2013; Muhala et al., 2024b), phylogenetic studies (Irwin et al.,

2019), and the monitoring and assessment of biodiversity (Loh et al., 2023; Roesma et al., 2023; Sodré et al., 2012; Valdez-Moreno et al., 2021).

However, in addition to genomic information, other data could be important for different types of analyses. For example, in studies that utilize the barcoding region to identify species (de Sousa et al., 2022; Fadli et al., 2021; Lopez-Vaamonde et al., 2021; Muhala et al., 2024b), the most updated taxonomy of the group in question and collection points or areas of occurrence (Sodré et al., 2012) could be useful.

Important data can be obtained through queries in the most commonly used repositories, which include the National Center for Biotechnology Information (NCBI) GenBank (Sayers et al., 2024), The

* Corresponding authors at: Laboratório de Evolução (LEVO), Instituto de Estudos Costeiros (IECOS), Universidade Federal do Pará (UFPA), Campus de Bragança, Bragança, Pará, Brazil.

E-mail addresses: luanrabelo@outlook.com (L. Rabelo), mvallino@ufpa.br (M. Vallinoto).

<https://doi.org/10.1016/j.ecolinf.2024.102970>

Received 19 June 2024; Received in revised form 17 December 2024; Accepted 17 December 2024

Available online 22 December 2024

1574-9541/© 2024 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Table 1
List of databases accessed by dataFishing through its public APIs.

Datasets	Data	dataFishing web form	dataFishing script
International Union for Conservation of Nature's Red List of Threatened Species (IUCN)	Common Names	YES	YES
	Country		
	Occurrence		
	Habitats		
	Status		
National Center for Biotechnology Information (NCBI GenBank)	Conservation	NO	YES
	Synonyms		
	Names		
The Barcode of Life Data Systems (BOLD Systems)	Taxonomy	NO	YES
	Sequences		
	Taxonomy		
	BINS		
	Sample IDs		
The Global Biodiversity Information Facility (GBIF)	Sequences	YES	YES
	Occurrence		
	Taxonomy		
	Authors		
	Basionym		
World Register of Marine Species (WoRMS)	Taxonomic	YES	YES
	Status		
	Taxonomy		
	Vernacular		
	Names		
	Aphia ID	YES	YES
	Authors		
	Species Status		
	Taxonomy		

Barcode of Life Data Systems (BOLD Systems) (Ratnasingham and Hebert, 2007), World Register of Marine Species (WoRMS) (Costello et al., 2013), Global Biodiversity Information Facility (GBIF) (Robertson et al., 2014), and the International Union for Conservation of Nature's Red List of Threatened Species (IUCN).

These databases require the species name to be provided to access the data; thus, the number of queries depends on the taxonomic group being studied, which can range from dozens to hundreds of queries. This compels researchers to conduct extensive searches in each database, making it a time-consuming and error-prone process. This problem can be resolved by using tools that connect to the databases' application programming interfaces (APIs) in workflows, which reduces the likelihood of errors and automates data access and processing, or by using tools developed for this purpose (Chamberlain, 2020; Chamberlain et al., 2022; Chamberlain and Szöcs, 2013).

Various R language scripts have been created to facilitate the mining and retrieval of genomic and taxonomic data from public databases. However, as these scripts have multiple functions and depend on the installation of various packages, researchers must have prior knowledge in writing commands, installing packages, executing command lines, and manipulating data to obtain the desired information. In addition, each script is specific to a function or a database; for example, using the R package *bold* (available at <https://github.com/ropensci/bold>), it is possible to obtain the taxonomy or sequence of a particular species. However, if the researcher also needs to obtain conservation status, they must use another R package, such as *rredlist* (Chamberlain, 2020) (available at <https://github.com/ropensci/rredlist>). Other useful R packages include *rgbif* (Chamberlain et al., 2022) for accessing data from the GBIF (available at <https://github.com/ropensci/rgbif>), *taxize* (Chamberlain and Szöcs, 2013) for taxonomic information across various sources (available at <https://github.com/ropensci/taxize>), and *worms* for accessing data from the WoRMS (available at <https://github.com/ropensci/worms>).

Additionally, the scripts listed above do not have a web interface, which could facilitate their use by researchers with little or no

Table 2
Overview of the dataFishing.py script's command-line arguments and associated databases for querying biological sequence data from platforms such as GenBank, BOLD Systems, IUCN, GBIF, and WoRMS.

Parameters	Descriptions
dataFishing.py	The Python script for <i>dataFishing</i> is available on GitHub: github.com/luanrabelo/dataFishing
-input {Mandatory} Path/String	A text file with species names listed on separate lines or a BOLD System TSV file.
-email {Mandatory ¹ } String	A previously registered email with NCBI using this link: https://account.ncbi.nlm.nih.gov/signup/ .
-download {Optional} boolean	Download sequence data from BOLD System and/or GenBank. Default is False.
-genesList {Mandatory ² } Path/String	A text file with genes names listed on separate lines. See Table 3 for more information.
-verbose {Optional} boolean	This argument specifies if the verbose mode should be enabled or disabled. Default is True.
-log {Optional} boolean	This argument specifies if the log mode should be enabled or disabled. Default is False.
-(databases) {Mandatory}	Select the database(s) you wish to query. The default is -all , which queries all the databases listed below. The databases and their parameters are available in the following list.
Databases	Parameters
International Union for Conservation of Nature's Red List of Threatened Species (IUCN)	-iucn True
National Center for Biotechnology Information (NCBI GenBank)	-ncbi True
The Barcode of Life Data Systems (BOLD Systems)	-bold True
The Global Biodiversity Information Facility (GBIF)	-gbif True
World Register of Marine Species (WoRMS)	-worms True

¹ if -all True or -ncbi True.
² if -download True.

experience in script execution. Therefore, this study introduces a new tool written in Python for mining, retrieving, and processing organelle DNA sequences and biodiversity information from various public databases (see Table 1). We present dataFishing, a Python tool available for free on GitHub at <https://github.com/luanrabelo/dataFishing>, under the MIT license, which ensures the freedom to use, modify, and distribute it.

This tool promotes searches in five databases (BOLD Systems, GBIF, GenBank, IUCN and WoRMS) and provides taxonomic and conservation information for a user-defined list of species. It is possible to add sequence data obtained from BOLD Systems and GenBank in a fast and easy way (see Results). Also, dataFishing provides different output files, which are not customizable, instead they are defined in function of the database used in a single search (i.e Genbank or WoRMS).

We have also created a web form, which can be accessed for free at <https://luanrabelo.github.io/dataFishing>, to enable researchers with little or no experience in developing or executing R or Python code to use some of the functions of dataFishing (see Table 1).

2. Materials and methods

dataFishing is a tool designed for organelle DNA sequences and biodiversity data mining, facilitating the querying, acquisition, and processing of these data. This tool can be executed in two ways: the first is through a webform (<https://luanrabelo.github.io/dataFishing>) where it is possible to obtain a file in Excel format with information about

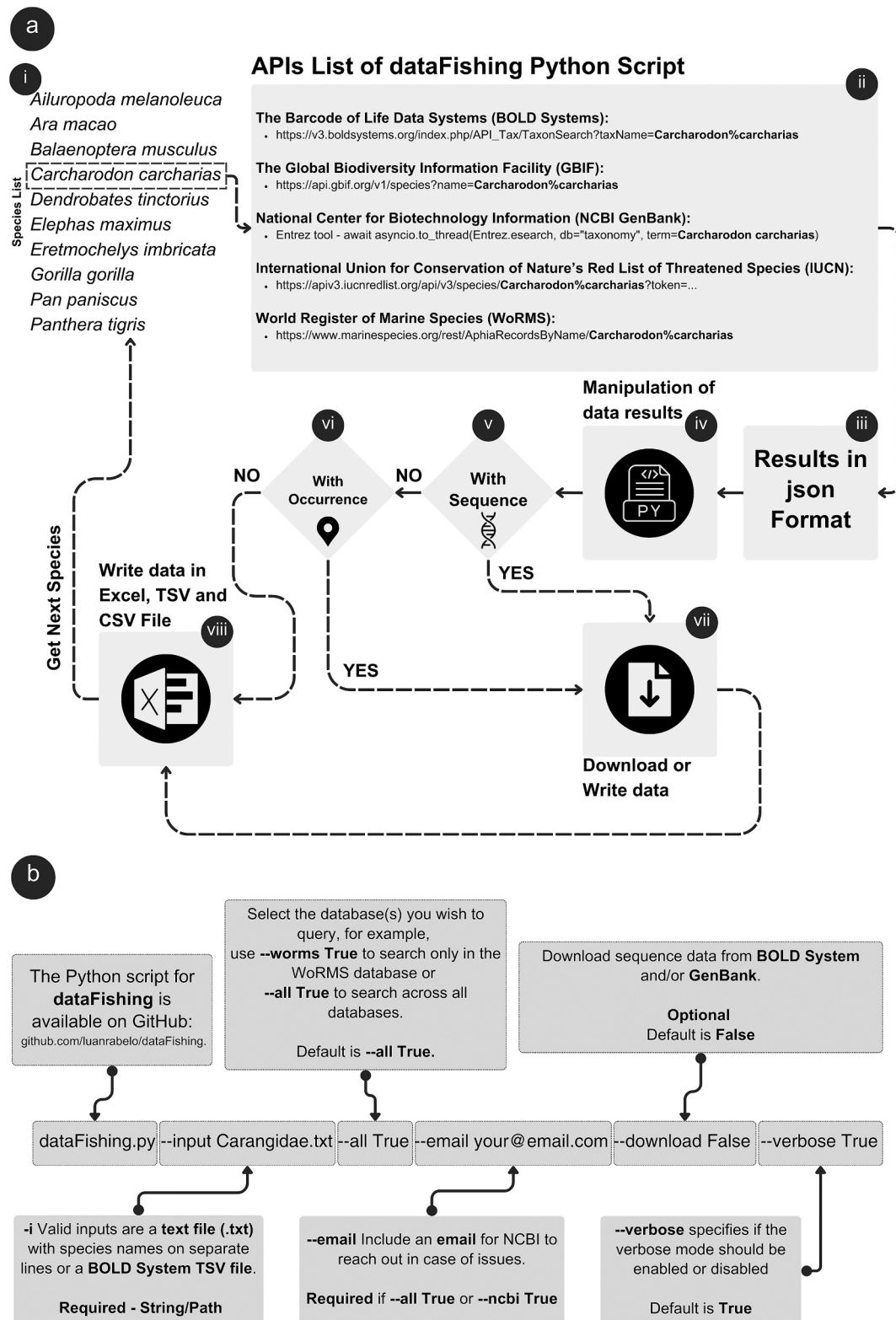


Fig. 1. (a) Workflow of the dataFishing tool, illustrating the process of mining mitochondrial and chloroplast sequences and biodiversity data from various public databases. (b) Command for executing the script; for more information, please refer to the documentation on GitHub.

taxonomy, conservation status, common names in different countries, and synonyms of species. All of these steps are performed quickly and intuitively and are recommended for researchers with little or no experience in running scripts or tools that use command lines. However, the execution of some queries using the web form is not possible due to

limitations imposed by the server. For example, BOLD Systems has CORS (Cross-Origin Resource Sharing) (Shaji and Subramanian, 2021), which is a security measure implemented by browsers to restrict how resources on a page can be requested from a domain different from that of the page itself. For more information, please refer to Table 1.

The second method is through the execution of a script, freely available at <https://github.com/luanrabelo/dataFishing>, under the MIT license. This script is written in Python and can be implemented in various workflows. It requires an active Python 3.12+ environment and an internet connection, which are recommended for researchers with advanced knowledge in executing scripts via the command line (see Table 2).

For the mining process using the Python script, some libraries need to be installed on the computer where the script will be executed, such as *requests* (<https://github.com/psf/requests>) and *aiohttp* (<https://github.com/aio-libs/aiohttp>), to perform queries using the available APIs (see Table 1); *SynGenes* (<https://github.com/luanrabelo/SynGenes>) (Rabelo et al., 2024), for generating commands used to search for sequences or manuscripts of mitochondrial or chloroplast markers, in the GenBank and PubMedCentral databases, respectively; *biopython* (<https://github.com/biopython/biopython>) (Cock et al., 2009), for reading and searching sequences, thus making it possible to obtain information contained in these data, such as the collection location; *pandas* (<https://github.com/pandas-dev/pandas>), for manipulating the data obtained through API queries or reading sequences data files; *openpyxl* (<https://github.com/theorchard/openpyxl>); and *xlsxwriter* (<https://github.com/jmcnamara/XlsxWriter>), for writing the information obtained from the databases into Excel format files.

The dataFishing script includes modules that check for the presence of the dataFishing libraries on the computer where it is being executed. If any libraries are missing, it prompts the user for permission with an installation message and then proceeds with an automatic installation. Additionally, dataFishing saves the query results in XLSX (Excel) CSV (Comma-Separated Values) and TSV (Tab-Separated Values) file formats. These files can be used in other scripts, both in R and Python, for data manipulation or by tools for visualizing geographic information, such as QGIS, available at <https://qgis.org>.

2.1. Genetic markers from organelles, taxonomic and biodiversity APIs

The APIs, of the databases presented in Table 1, are used to obtain information on taxonomy, collection sites, conservation status, and occurrences, among others. Most of these APIs allow for asynchronous searching, which facilitates multiple simultaneous searches in each database, thereby reducing the computational time. However, others permit only one query at a time, such as GenBank.

To prevent server overload, GenBank recommends that tools used for searches in this database limit their queries to a maximum of three per second or up to ten if the user has an API key (<https://www.ncbi.nlm.nih.gov/books/NBK25497/>). dataFishing adheres to this guideline and adds a waiting time between queries on GenBank, avoiding temporary blocks, but this increases the computational time for performing functions that depend on a specific database (see Results). A workflow of the operation of the API queries is shown in Fig. 1a.

2.2. Input and output data

The dataFishing script works based on keyword searching, requiring a single species name or a list of unique species names as a parameter (see Fig. 1b and Table 2) to initiate the mining process and obtain the taxonomic data. It is important to mention that it is not possible to obtain the results from different names (synonymous) of a single species in a single search. However, users can generate synonym lists from sources such as *taxize*, *rredlist*, and *rgbif* and use those as queries to expand their searches.

This keyword can be a simple text file with species names separated by lines, without any special characters, or a TSV file, which can be obtained previously from the BOLD Systems database by searching for a higher-level taxonomic group, such as an order, family, or genus (see Supplementary Materials 1–10).

If the user also wishes to obtain sequence data from GenBank, they

Table 3
List of mitochondrial and chloroplast genes accepted by dataFishing for sequence searches.

Mitochondrial Genes	Codes
rRNAs	12S; 16S
Mitochondrial Complex I	ND1; ND2; ND3; ND4; ND4L; ND5; ND6
Mitochondrial Complex III	CYTB
Mitochondrial Complex IV	COI; COII; COIII
Mitochondrial Complex V	ATP6; ATP8
Control Region	Control Region
Chloroplast Genes	Codes
rRNAs	rrn16S; rrn23S; rrn4.5S; rrn5S
ATP Synthase	atpA; atpB; atpE; atpF; atpH; atpI
Cytochrome b/6f Complex	petA; petB; petD; petE; petG; petL; petN
DNA dependent RNA polymerase	rpoA; rpoB; rpoC1; rpoC2
Large Subunit of Ribosome	rpl2; rpl14; rpl16; rpl20; rpl22; rpl23; rpl32; rpl33; rpl36
NADH-dehydrogenase	ndhA; ndhB; ndhC; ndhD; ndhE; ndhF; ndhG; ndhH; ndhI; ndhJ; ndhK
PhotoSystem I-II	psaA; psaB; psaC; psal; psaJ; psam; psb30; psbA; psbB; psbC; psbD; psbE; psbF; psbH; psbI; psbJ; psbK; psbL; psbM; psbN; psbZ
Small Subunit of Ribosome	rps2; rps3; rps4; rps7; rps8; rps11; rps12; rps14; rps15; rps16; rps18; rps19
Rubisco	rbcl
Others	accD; ccsA; cemA; chlB; chlL; chlN; clpP; clpP1; cysA; cysT; ftsH; infA; lhbA; matK; pafI; pafII; pbfI; psb30; ycf1; ycf2; ycf3; ycf4; ycf12; ycf15;

must provide a list of genes in a text file (see Table 3). It is important to mention that dataFishing will perform searches over the entire GenBank accession file.

The list of genes accepted by dataFishing (Table 3) is detailed in its documentation on its GitHub repository, which also contains example files for gene names for animals and plants.

Regarding the TSV file obtained previously from BOLD Systems, dataFishing could verify all the species names in this file to eliminate redundancies, thus preventing the script from conducting multiple searches for the same species and reducing computational time. Subsequently, dataFishing will create folders to organize the data in the TSV file. For example, if the user has a file with data from the Carangidae (Pisces) family, dataFishing will automatically create separate folders for each genus within the family. Similarly, it will also create folders for each species within those genera folders and store the corresponding sequences in their respective FASTA files.

The taxonomic data, Barcode Index Numbers (BINs), geographic coordinates, specimen vouchers, GenBank vouchers, country, and region will be collected from the TSV file, and they could be inserted into an Excel file.

The output of dataFishing has the same information obtained by other tools, but better organized in XLSX, TSV and CSV files. It is important to mention that, as *SynGenes* library has been included in the dataFishing, the results from GenBank searches using dataFishing could provide better results when compared to traditional GenBank searches.

3. Results

3.1. dataFishing – web form

The dataFishing web form (Fig. 2) possesses various functionalities for obtaining information from the databases listed in Table 1. It is an easy task through the use of dataFishing to obtain an Excel file containing taxonomic information, occurrence data, conservation status, and common names in different countries, among others features. However, the dataFishing web form cannot access all the databases

1 Select the data source to search for biodiversity information:

☒ International Union for Conservation of Nature's Red List of Threatened Species (IUCN) ☐ Global Biodiversity Information Facility (GBIF) ☐ World Register of Marine Species (WORMS)

Search Options of International Union for Conservation of Nature's Red List of Threatened Species (IUCN)

☐ All data ☒ Common Names ☐ Country Occurrence ☐ Habitats ☒ Species Author* ☒ Status Conservation* ☐ Synonyms Names ☒ Taxonomy*

*** Mandatory Fields**

1 Click and select the options to search for International Union for Conservation of Nature's Red List of Threatened Species (IUCN)

2 Please note that a full data search may take considerable time. We appreciate your patience during processing.

Species Names:

Ara macao
Balaenoptera musculus
Carcharodon carcharias
Dendrobates tinctorius
Elephas maximus
Eretmochelys imbricata
Gorilla gorilla
Pan paniscus
Panthera tigris

Enter the Species Names, each on a new line, without any special characters. [Click here](#) to see an example.

Start dataFishing

2 Visualize and export the results

Export Result data to Excel

Kingdom	Phylum	Class	Order	Family	Genus	Species	Common Names	Country Occurrence	Habitats	Species Citation	Status Conservation	Synonyms Names	Link
Animalia	Chordata	Mammalia	Artiodactyla	Balaenopteridae	Balaenoptera	Balaenoptera musculus (Linnaeus, 1758)	Blue Whale ^{eng} Baleine bleue ^{fre} Ballena Azul ^{spa}	-	-	(Linnaeus, 1758)	Endangered (EN)	Balaena musculus Linnaeus, 1758	Link
Animalia	Chordata	Aves	Psittaciformes	Psittacidae	Ara	Ara macao (Linnaeus, 1758)	Scarlet Macaw ^{eng}	-	-	(Linnaeus, 1758)	Least Concern (LC)	-	Link
Animalia	Chordata	Reptilia	Testudines	Cheloniidae	Eretmochelys	Eretmochelys imbricata (Linnaeus, 1766)	Hawksbill Turtle ^{eng} Tortue imbriquée ^{fre} Tortue à écailles ^{fre} Tortue à bec faucon ^{fre} Tortue Caret ^{fre} Caret ^{fre} Tortuga de Carey ^{spa}	-	-	(Linnaeus, 1766)	Critically Endangered (CR)	Testudo imbricata Linnaeus, 1766	Link
Animalia	Chordata	Mammalia	Primates	Hominidae	Gorilla	Gorilla gorilla (Savage, 1847)	Western Gorilla ^{eng} Lowland Gorilla ^{eng} Gorille de l'Ouest ^{fre} Gorilla Occidental ^{spa}	-	-	(Savage, 1847)	Critically Endangered (CR)	Troglodytes gorilla Savage, 1847 Gorilla gorilla Gorilla gorilla Gorilla gorilla	Link
Animalia	Chordata	Mammalia	Primates	Hominidae	Pan	Pan paniscus Schwarz, 1929	Bonobo ^{eng} Pygmy Chimpanzee ^{eng} Gracile Chimpanzee ^{eng} Dwarf Chimpanzee ^{eng} Bonobo ^{fre} Bonobo ^{spa}	-	-	Schwarz, 1929	Endangered (EN)	Pan satyrus	Link
						Dyeing Poison Frog ^{eng} Dendrobates	Tinging Frog ^{eng}					Dendrobates tinctorius Calamita tinctorius (Cuvier, 1797) Dendrobates azureus Hoogmoed, 1969 Dendrobates machadoi Bokermann, 1958	

Fig. 2. dataFishing Web Form, available at <https://luanrabelo.github.io/dataFishing>. (a) Select one of the available databases to search for information. (b) Choose the data you wish to retrieve from the databases; note that some data fields are mandatory. (c) Enter the names of the species, using only standard characters, to obtain the previously selected information. An example button is provided for demonstration purposes. (d) View the results, with the option to export them to an Excel file.

listed in Table 1 (see the Cross-Origin Resource Sharing explained before). Therefore, to access information from all the databases, it is necessary to run the Python script.

3.2. dataFishing - Python Script

The dataFishing is a script that runs on Windows, Linux and MacOS equally. To present the functionality of dataFishing script, we used 2494 species names from 10 different families, including reptiles, amphibians,

Table 4
Number of species per family used in the dataFishing analyses. The data were retrieved from BOLD Systems in March 2024.

Order	Family	Species Count	Supplementary Material
Anura	Bufonidae	288 Species	Supplementary Material 1
Perciformes	Carangidae	156 Species	Supplementary Material 2
Carcharhiniformes	Carcharhinidae	76 Species	Supplementary Material 3
Primates	Cercopithecidae	131 Species	Supplementary Material 4
Clupeiformes	Clupeidae	214 Species	Supplementary Material 5
Columbiformes	Columbidae	184 Species	Supplementary Material 6
Squamata	Lacertidae	165 Species	Supplementary Material 7
Siluriformes	Loricariidae	628 Species	Supplementary Material 8
Perciformes	Sciaenidae	216 Species	Supplementary Material 9
Chiroptera	Vespertilionidae	436 Species	Supplementary Material 10
Total:		2494 Species	

fishes, avians and mammals. The analyses were conducted using a computer with an AMD Ryzen 7 Series 5000 processor, 16 GB of RAM, and a 300 Mbps internet connection, where time values may vary with different configurations. The R scripts used in the comparative analyses below can be found in the dataFishing repository on GitHub.

To obtain the species names that served as the basis for the searches, we conducted searches in the BOLD Systems using the family name (Table 4) and subsequently obtained a TSV file with the results (Supplementary Material 1–10). We then eliminated duplicates and retained only the unique species names to avoid multiple queries on the same species, thus consuming less computational time. We used *pandas* in Python and the *tidyverse* package in R to manipulate the files.

These families (Table 4) were selected to represent a variable proportion of threatened species. For instance, the Cercopithecidae family has approximately 72.5 % of its species under some degree of threat (NT, VU, EN, CR), while Carcharhinidae has 68 %, according to the IUCN in 2024.

3.3. Comparative analysis of taxonomic data retrieval

In this study, we evaluated the performance of several R packages from the rOpenSci project (*bold*, *taxize*, *rgbif*, *rredlist*, and *worms*) and compared them to the dataFishing tool for retrieving taxonomic data across various species families (Table 4). A total of 2494 API queries were conducted to assess how efficiently each tool performed. Throughout the analyses, dataFishing consistently demonstrated a clear advantage over the other tools in terms of speed.

For the BOLD Systems, the *bold* and *taxize* packages were used to retrieve taxonomic data for species from several families. For instance, retrieving data, on average, across all families tested (Table 4), dataFishing significantly outperformed both *taxize* and *bold*. The average retrieval time was 49.2 s with dataFishing, compared to 94.7 s with *taxize* and 89.4 s with *bold*. These results highlight the efficiency of dataFishing, with it being approximately 1.9× faster than *taxize* and 1.8× faster than *bold* on average (see Fig. 3a).

When searching the NCBI GenBank, retrieving taxonomy data from all the families tested (Table 4), dataFishing showed consistent efficiency, with lower variability across taxonomic groups. Its average retrieval time was 134 s, compared to 264 s for *taxize*, highlighting dataFishing’s speed and reliability, even for larger datasets (see Fig. 3b).

In retrieving data from the GBIF, both the *rgbif* and *taxize* packages required two steps, retrieving species IDs and then obtaining taxonomy. On average, dataFishing completed tasks in 8.4 s, compared to 152.3 s with *taxize* and 182 s with *rgbif*. Its efficiency was consistent across families, including larger datasets like Loricariidae (see Table 4), where dataFishing significantly outperformed the alternatives (see Fig. 3c).

For taxonomic data from the IUCN Red List, the *rredlist* and *taxize* packages were used. On average, dataFishing completed tasks in 36.3 s, compared to 153.9 s with *taxize* and 152.2 s with *rredlist* (see Fig. 3d).

Finally, when retrieving data from the WoRMS, *worms* required 212 s to retrieve taxonomic data for the Carangidae family, while dataFishing completed the task in just 3 s, making it approximately 70× faster. Similar time savings were observed for the Carcharhinidae, Clupeidae, Loricariidae, and Sciaenidae families, where dataFishing averaged 6.3 s, outperforming *worms* (347 s) (see Fig. 3e).

The data containing the execution times for each analysis, used to compare the performance of the tools dataFishing, *bold*, *taxize*, *rgbif*, *rredlist*, and *worms* in retrieving taxonomic information, are available in the GitHub repository.

3.4. Sequences, occurrence, and conservation

In addition to taxonomic data, dataFishing can also retrieve DNA sequences from the BOLD Systems (*COI*) and GenBank databases (various markers, mitochondrial or chloroplast) and store them in FASTA file format. For searches in GenBank, dataFishing uses commands generated by the Python class *SynGenes* (Rabelo et al., 2024) (refer to the *SynGenes* documentation for more details). The time taken to search and retrieve sequences through dataFishing in these databases is demonstrated in Fig. 4a, where in GenBank, 2467 *COI* sequences were obtained for species of the Bufonidae family; 6561 for Carangidae, 5289 for Carcharhinidae, 662 for Cercopithecidae, 4525 for Clupeidae, 1148 for Columbidae, 1174 for Lacertidae, 3138 for Loricariidae, 4602 for Sciaenidae, and 6256 for Vespertilionidae. It is worth noting that this function, following NCBI recommendations, is not performed asynchronously, thus increasing processing time. All GenBank sequence searches were performed in March 2024. Finally, sequences from BOLD Systems could be easily obtained from the TSV file, which it was previously obtained from the dataFishing search.

To obtain occurrence data from the BOLD Systems and/or GenBank metadata, once the sequence data are acquired, they will be read using the *biopython* library or *pandas*. This process includes checking for the presence of coordinate information, and if available, storing it. Fig. 4b illustrates the time required to obtain this information, including occurrence and sequences, from the input files of BOLD Systems and subsequent storage in TSV and FASTA files.

Fig. 4c presents the processing time needed to obtain various information from the IUCN. These information encompass the conservation status, which provides insight into the vulnerability and risk of extinction of species; common names, which facilitate the identification and communication about species in different languages and regions; countries of occurrence, indicating the geographical distribution of species and potentially aiding in understanding migration patterns and preferred habitats; habitats, offering insights into the natural environments where species thrive; and synonyms, which are essential for tracking changes in species nomenclature over time. This analysis utilized species from the families listed in Table 4.

4. Discussion

Using R scripts for mining genomic and taxonomic data from public databases is complex, requiring researchers to have skills in programming and data manipulation (Xiong et al., 2022). The specificity of scripts for functions or individual databases, such as the *bold* package for taxonomy, sequences, and *rredlist* for conservation status, increases the need for technical knowledge for proper execution.

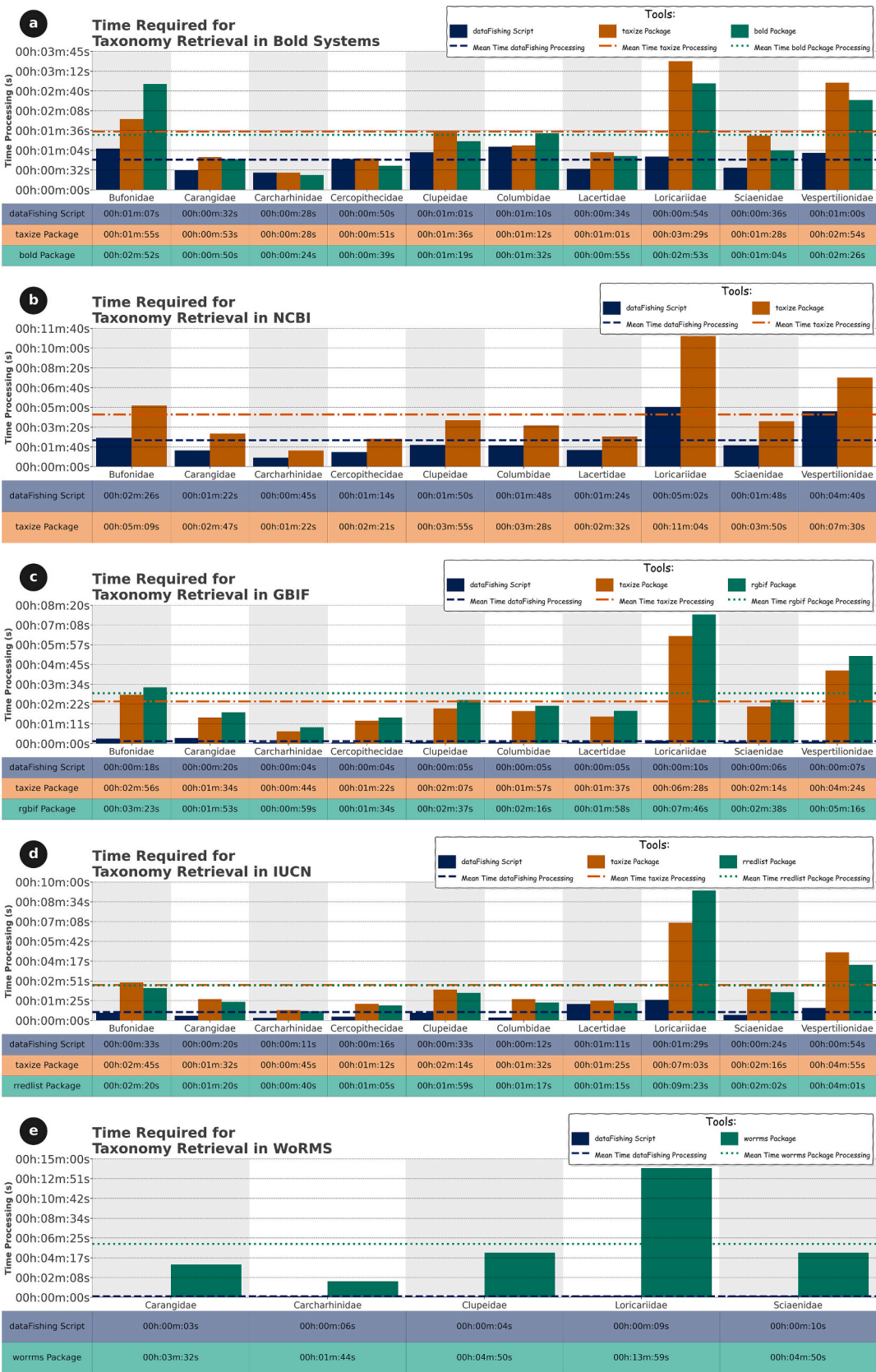


Fig. 3. Comparison of the efficiency of the dataFishing tool with that of other tools in obtaining taxonomic information from different databases, including BOLD Systems (a), GenBank (b), GBIF (c), IUCN (d), and WoRMS (e).

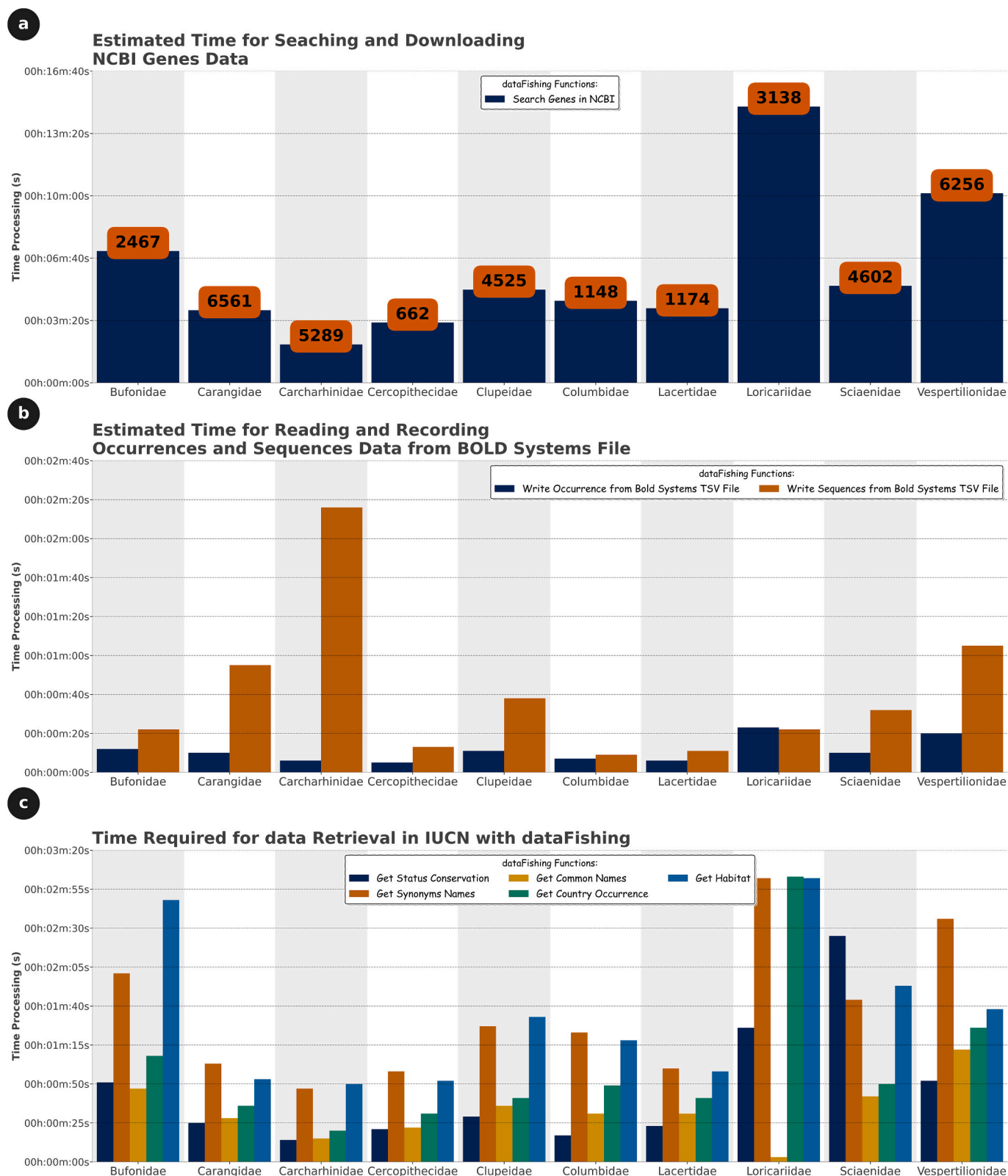


Fig. 4. (a) Efficiency of dataFishing in querying and retrieving sequences from GenBank. (b) Estimated time to obtain occurrence data and sequences from TSV files originating from BOLD Systems. (c) Time required to obtain various data from the IUCN.

dataFishing is a Python-developed tool that simplifies and automates the retrieval of organelle-specific genetic data, taxonomic, nomenclatural, and conservation status data. dataFishing allows access to various public databases through their APIs, enabling the acquisition of different

types of data from each database (see Table 1). In addition to being a script, this tool features a web form that facilitates the search and subsequent storage of information in an Excel file format. With a script and an intuitive web interface, it allows for simplified searches and direct

storage of information in Excel.

Therefore, compared to other tools, dataFishing offers a quick and practical solution for mining sequences from organelle genomes and biodiversity data and can be useful for various studies in biology, ecology, evolution, and conservation (Cardenosa et al., 2021; Muhala et al., 2024a; Muhala et al., 2024b; Vaini et al., 2021). The tool enables the acquisition of relevant information about species of interest, such as their taxonomy, conservation status, common names, synonyms, and occurrence points, from different online data sources, including The BOLD Systems, NCBI GenBank, GBIF, WoRMS, and IUCN.

The tool also facilitates the processing and analysis of these data, generating files in Excel and TSV formats, which can be used for various purposes, such as producing occurrence maps with the obtained coordinates or serving as input files for further analyses using other tools (Eller et al., 2019; Palacio et al., 2021).

Another convenience that can be highlighted is that the tool has two modes of use: one through a simpler and more intuitive web form and the other through a more flexible and customizable Python script. Both modes require only an internet connection, with the script also requiring an active Python 3.12+ environment. The dataFishing tool is, therefore, a valuable resource for the scientific community and can benefit from a fast and efficient way to mine genetic markers from mitochondria and chloroplasts and biodiversity data from different online data sources.

dataFishing can be improved with constant updates. In addition to making the source code open to the scientific community, we plan to include more APIs in future versions. Finally, we emphasize that requests for the inclusion of any specific API (or even errors) in the tool can be made through the dataFishing page on GitHub, thus socializing the construction of future versions.

Funding

This study was financed in part by CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) through a graduate scholarship to LPR and research productivity (PQ) fellowships to MV (303889/2022-5).

CRediT authorship contribution statement

Luan Rabelo: Writing – review & editing, Writing – original draft, Software. **Davidson Sodré:** Writing – review & editing, Software. **Oscar David Albito Balcázar:** Software. **Murilo Furtado do Rosário:** Software. **Aurycéia Jaquelyne Guimarães-Costa:** Writing – review & editing, Conceptualization. **Grazielle Gomes:** Writing – review & editing, Writing – original draft. **Iracilda Sampaio:** Writing – review & editing, Conceptualization. **Marcelo Vallinoto:** Writing – review & editing, Writing – original draft, Supervision, Funding acquisition, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the author **Luan Rabelo** used the AI Copilot from Microsoft to improve and correct the grammar of the text and the AI Curie from American Journal Experts to improve the structure of the text. After using these tools/services, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Marcelo Vallinoto reports financial support was provided by Federal University of Para. If there are other authors, they declare that they have no known competing financial interests or personal relationships that

could have appeared to influence the work reported in this paper.

Data availability

All data and code for this work are available on the dataFishing GitHub repository (<https://github.com/luanrabelo/dataFishing>).

Acknowledgments

We express our gratitude to the PAPQ of PROPESP/UFPA and the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) for their financial contributions.

Appendix A. Supplementary data

Supplementary data to this article can be found online at <https://doi.org/10.1016/j.ecoinf.2024.102970>.

References

- Bucklin, A., Peijnenburg, K.T.C.A., Kosobokova, K.N., O'Brien, T.D., Blanco-Bercial, L., Cornils, A., Falkenhaus, T., Hopcroft, R.R., Hosia, A., Laakmann, S., Li, C., Martell, L., Questel, J.M., Wall-Palmer, D., Wang, M., Wiebe, P.H., Weymann-Zwolicka, A., 2021. Toward a global reference database of COI barcodes for marine zooplankton. *Mar. Biol.* 168, 78. <https://doi.org/10.1007/s00227-021-03887-y>.
- Cardenosa, D., Fields, A.T., Shea, S.K.H., Feldheim, K.A., Chapman, D.D., 2021. Relative contribution to the shark fin trade of Indo-Pacific and eastern Pacific pelagic thresher sharks. *Anim. Conserv.* 24, 367–372. <https://doi.org/10.1111/acv.12644>.
- Chamberlain, S., 2020. Rredlist: "IUCN" Red List Client. R package version 0.7. 0. (See <https://CRAN.R-project.org/package=rredlist>).
- Chamberlain, S.A., Szöcs, E., 2013. Taxize: taxonomic search and retrieval in R. *F1000Res* 2, 191. <https://doi.org/10.12688/f1000research.2-191.v2>.
- Chamberlain, S., Oldoni, D., Waller, J., 2022. Rgbif: interface to the global biodiversity information facility API.
- Cock, P.J.A., Antao, T., Chang, J.T., Chapman, B.A., Cox, C.J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., De Hoon, M.J.L., 2009. Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* 25, 1422–1423. <https://doi.org/10.1093/bioinformatics/btp163>.
- Costello, M.J., Bouchet, P., Boxshall, G., Fauchald, K., Gordon, D., Hoeksema, B.W., Poore, G.C.B., van Soest, R.W.M., Störh, S., Walter, T.C., Vanhoorne, B., Decock, W., Appeltans, W., 2013. Global coordination and standardisation in marine biodiversity through the world register of marine species (WoRMS) and related databases. *PLoS One* 8, e51629. <https://doi.org/10.1371/journal.pone.0051629>.
- da Costa, K.G., Bezerra, T.R., Monteiro, M.C., Vallinoto, M., Berrêdo, J.F., Pereira, L.C.C., da Costa, R.M., 2013. Tidal-induced changes in the zooplankton Community of an Amazon Estuary. *J. Coast. Res.* 289, 756–765. <https://doi.org/10.2112/JCOASTRES-D-11-00166.1>.
- de Sousa, R.P.C., Bessa-Brito, C.D., Guimarães-Costa, A., Evangelista-Gomes, G., Sampaio, I., de Oliveira, E.H.C., Vallinoto, M., 2022. Exploring the diversity of Elopidae (Teleostei: Elopiformes) using DNA barcoding analysis. *Diversity* (Basel) 14, 1008. <https://doi.org/10.3390/d14111008>.
- Dias, C., de Lima, K.A., Araripe, J., Aleixo, A., Vallinoto, M., Sampaio, I., Schneider, H., do Rêgo, P.S., 2018. Mitochondrial introgression obscures phylogenetic relationships among manakins of the genus *Lepidothrix* (Aves: Pipridae). *Mol. Phylogenet. Evol.* 126, 314–320. <https://doi.org/10.1016/j.ympev.2018.04.017>.
- Eller, R.J., Janga, S.C., Walsh, S., 2019. Odyssey: a semi-automated pipeline for phasing, imputation, and analysis of genome-wide genetic data. *BMC Bioinformatics* 20, 364. <https://doi.org/10.1186/s12859-019-2964-5>.
- Fadli, N., Muchlisin, Z.A., Siti-Azizah, M.N., 2021. DNA barcoding of commercially important groupers (Epinephelidae) in Aceh, Indonesia. *Fish. Res.* 234, 105796. <https://doi.org/10.1016/j.fishres.2020.105796>.
- Irwin, N.A.T., Tikhonenkov, D.V., Hehenberger, E., Mylnikov, A.P., Burki, F., Keeling, P. J., 2019. Phylogenomics supports the monophyly of the Cercozoa. *Mol. Phylogenet. Evol.* 130, 416–423. <https://doi.org/10.1016/j.ympev.2018.09.004>.
- Ito, R.K., Harada, S., Tabata, R., Watanabe, K., 2022. Molecular evolution and convergence of the rhodopsin gene in *Gymnogobius*, a goby group having diverged into coastal to freshwater habitats. *J. Evol. Biol.* 35, 333–346. <https://doi.org/10.1111/jeb.13955>.
- Loh, K.-H., Lim, K.-C., Then, A.Y.-H., Adam, S., Leung, A.J.-X., Hu, W., Bong, C.W., Wang, A., Sade, A., Musel, J., Du, J., 2023. Advancing DNA barcoding to elucidate elasmobranch biodiversity in Malaysian waters. *Animals* 13, 1002. <https://doi.org/10.3390/ani13061002>.
- Lopez-Vaamonde, C., Kirichenko, N., Cama, A., Doorenweerd, C., Godfray, H.C.J., Guiguet, A., Gomboc, S., Huemer, P., Landry, J.-F., Laštůvka, A., Laštůvka, Z., Lee, K. M., Lees, D.C., Mutanen, M., van Nieukerken, E.J., Segerer, A.H., Triberti, P., Wieser, C., Rougerie, R., 2021. Evaluating DNA barcoding for species identification and discovery in European Gracillariid moths. *Front. Ecol. Evol.* 9. <https://doi.org/10.3389/fevo.2021.626752>.
- Louro, M., Kuzmina, T.A., Bredtmann, C.M., Diekmann, I., de Carvalho, L.M.M., von Samson-Himmelstjerna, G., Krücken, J., 2021. Genetic variability, cryptic species

- and phylogenetic relationship of six cyathostomin species based on mitochondrial and nuclear sequences. *Sci. Rep.* 11, 8245. <https://doi.org/10.1038/s41598-021-87500-8>.
- Muhala, V., Guimarães-Costa, A., Bessa-Silva, A.R., Rabelo, L.P., Carneiro, J., Macate, I. E., Watanabe, L., Balcázar, O.D., Gomes, G.E., Vallinoto, M., Sampaio, I., 2024a. Comparative mitochondrial genome brings insights to slight variation in gene proportion and large intergenic spacer and phylogenetic relationship of mudskipper species. *Sci. Rep.* 14, 3358. <https://doi.org/10.1038/s41598-024-52979-4>.
- Muhala, V., Guimarães-Costa, A., Macate, I.E., Rabelo, L.P., Bessa-Silva, A.R., Watanabe, L., dos Santos, G.D., Sambora, L., Vallinoto, M., Sampaio, I., 2024b. DNA barcoding for the assessment of marine and coastal fish diversity from the coast of Mozambique. *PLoS One* 19, e0293345. <https://doi.org/10.1371/journal.pone.0293345>.
- Nakazato, T., 2020. Survey of species covered by DNA barcoding data in BOLD and GenBank for integration of data for Museomics. *Biodivers. Inform. Sci. Stand.* 4. <https://doi.org/10.3897/biss.4.59065>.
- Palacio, R.D., Negret, P.J., Velásquez-Tibatá, J., Jacobson, A.P., 2021. A data-driven geospatial workflow to map species distributions for conservation assessments. *Divers. Distrib.* 27, 2559–2570. <https://doi.org/10.1111/ddi.13424>.
- Rabelo, L.P., Sodré, D., de Sousa, R.P.C., Watanabe, L., Gomes, G., Sampaio, I., Vallinoto, M., 2024. SynGenes: a Python class for standardizing nomenclatures of mitochondrial and chloroplast genes and a web form for enhancing searches for evolutionary analyses. *BMC Bioinformatics* 25, 160. <https://doi.org/10.1186/s12859-024-05781-y>.
- Ratnasingham, S., Hebert, P.D.N., 2007. BOLD: the barcode of life data system: barcoding. *Mol. Ecol. Notes* 7, 355–364. <https://doi.org/10.1111/J.1471-8286.2007.01678.X>.
- Robertson, T., Döring, M., Guralnick, R., Bloom, D., Wiczeorek, J., Braak, K., Otegui, J., Russell, L., Desmet, P., 2014. The GBIF integrated publishing toolkit: facilitating the efficient publishing of biodiversity data on the internet. *PLoS One* 9, e102623. <https://doi.org/10.1371/journal.pone.0102623>.
- Rocha, T.C., Sequeira, F., Aleixo, A., Rêgo, P.S., Sampaio, I., Schneider, H., Vallinoto, M., 2015. Molecular phylogeny and diversification of a widespread Neotropical rainforest bird group: the buff-throated Woodcreeper complex, *Xiphorhynchus guttatus/susurrans* (Aves: Dendrocolaptidae). *Mol. Phylogenet. Evol.* 85, 131–140. <https://doi.org/10.1016/j.ympev.2015.02.004>.
- Roesma, D.I., Tjong, D.H., Syaifullah, S., Nofrita, N., Janra, M.N., Prawira, F.D.L., Salid, V.M., Aidil, D.R., 2023. The importance of DNA barcode reference libraries and selection primer pair in monitoring fish diversity using environmental DNA metabarcoding. *Biodiversitas* 24. <https://doi.org/10.13057/biodiv/d240438>.
- Sayers, E.W., Cavanaugh, M., Clark, K., Pruitt, K.D., Sherry, S.T., Yankie, L., Karsch-Mizrachi, I., 2024. GenBank 2024 Update. *Nucleic Acids Res.* 52, D134–D137. <https://doi.org/10.1093/nar/gkad903>.
- Sequeira, F., Bessa-Silva, A., Tarroso, P., Sousa-Neves, T., Vallinoto, M., Gonçalves, H., Martínez-Solano, I., 2020. Discordant patterns of introgression across a narrow hybrid zone between two cryptic lineages of an Iberian endemic newt. *J. Evol. Biol.* 33, 202–216. <https://doi.org/10.1111/jeb.13562>.
- Shaji, E., Subramanian, N., 2021. Assessing non-intrusive vulnerability scanning methodologies for detecting web application vulnerabilities on large scale. In: 2021 International Conference on System, Computation, Automation and Networking (ICSCAN). IEEE, pp. 1–5. <https://doi.org/10.1109/ICSCAN53069.2021.9526423>.
- Sodré, D., Rodrigues-Filho, L.F.S., Souza, R.F.C., Rêgo, P.S., Schneider, H., Sampaio, I., Vallinoto, M., 2012. Inclusion of South American samples reveals new population structuring of the blacktip shark (*Carcharhinus limbatus*) in the western Atlantic. *Genet. Mol. Biol.* 35, 752–760. <https://doi.org/10.1590/S1415-47572012005000062>.
- Song, W., Ji, C., Chen, Z., Cai, H., Wu, X., Shi, C., Wang, S., 2022. Comparative analysis the complete chloroplast genomes of nine *Musa* species: genomic features, comparative analysis, and phylogenetic implications. *Front. Plant Sci.* 13. <https://doi.org/10.3389/fpls.2022.832884>.
- Vaini, J.O., Domingues, R.R., da Ferrette, B.L.S., Hallerman, E.M., Mota, K.G., Barreiros, J.P., Hilsdorf, A.W.S., 2021. Global population genetic structure of the sequential hermaphrodite, dusky grouper (*Epinephelus marginatus*). *Aquat. Conserv.* 31, 2119–2130. <https://doi.org/10.1002/aqc.3608>.
- Valdez-Moreno, M., Mendoza-Carranza, M., Rendón-Hernández, E., Alarcón-Chavira, E., Elías-Gutiérrez, M., 2021. DNA barcodes applied to a rapid baseline construction in biodiversity monitoring for the conservation of aquatic ecosystems in the Sian Ka'an reserve (Mexico) and adjacent areas. *Diversity (Basel)* 13, 292. <https://doi.org/10.3390/d13070292>.
- Vallinoto, M., Schneider, M.P.C., Silva, A., Iannuzzi, L., Brenig, B., 2004. Molecular cloning and analysis of the swamp and river buffalo leptin gene. *Anim. Genet.* 35, 462–463. <https://doi.org/10.1111/j.1365-2052.2004.01186.x>.
- Xiong, K., Luo, Z., Fu, S., Wang, Y., Xu, M., Wu, Y., 2022. Revealing the semantics of data wrangling scripts with COMANTICS. *IEEE Trans. Vis. Comput. Graph.* 1–11. <https://doi.org/10.1109/TVCG.2022.3209470>.
- Xu, S., Teng, K., Zhang, Hui, Wu, J., Duan, L., Zhang, Hongyu, Wen, H., Teng, W., Yue, Y., Fan, X., 2023. The first complete mitochondrial genome of *Carex* (*C. breviculmis*): a significantly expanded genome with highly structural variations. *Planta* 258, 43. <https://doi.org/10.1007/s00425-023-04169-1>.