Received: 13 October 2023 | Revised: 23 March 2024 | Accepted: 27 March 2024

DOI: 10.1111/geb.13847

Global Ecology and Biogeography A Journal of Macroecology WILEY

METHOD

occTest: An integrated approach for quality control of species occurrence data

Josep M. Serra-Diaz[1,2] | Jeremy Borderieux[2] | Brian Maitner[3] | Coline C. F. Boonman[4] | Daniel Park[5,6] | Wen-Yong Guo[7] | Arnaud Callebaut[2] | Brian J. Enquist[8,9] | Jens-C. Svenning[4] | Cory Merow[1]

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

[2]AgroParisTech, INRAE Silva, Université de Lorraine, Nancy, France

[3]Department of Geography, University at Buffalo, Buffalo, New York, USA

[4]Department of Biology, Center for Ecological Dynamics in a Novel Biosphere (ECONOVO) & Center for Biodiversity Dynamics in a Changing World (BIOCHANGE), Department of Biology, Aarhus University, Aarhus C, Denmark

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

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

[7]Research Center for Global Change and Complex Ecosystems & Zhejiang Tiantong Forest Ecosystem National Observation and Research Station, School of Ecological and Environmental Sciences, East China Normal University, Shanghai, P.R. China

[8]Department of Ecology and Evolutionary Biology, University of Arizona, Tucson, Arizona, USA

[9]Santa Fe Institute, Santa Fe, New Mexico, USA

Correspondence
Josep M. Serra-Diaz, Department of Ecology and Evolutionary Biology and Eversource Energy Center, University of Connecticut, 75 N. Eagleville Road, Unit 3043, Storrs, CT 06269-3043, USA.
Email: pep.bioalerts@gmail.com

Funding information
Agence Nationale de la Recherche, Grant/Award Number: ANR-21-CE32-0003; Danmarks Grundforskningsfond, Grant/Award Number: DNRF173; Advanced Exploration Systems, Grant/Award Number: NASA Grant 80NSSC22K0883; Conseil régional du Grand Est, Grant/Award Number: 19_GE8_01020p05035; Villum Fonden, Grant/Award Number: 16549

Handling Editor: Andres Baselga

Abstract

Aim: Species occurrence data are valuable information that enables one to estimate geographical distributions, characterize niches and their evolution, and guide spatial conservation planning. Rapid increases in species occurrence data stem from increasing digitization and aggregation efforts, and citizen science initiatives. However, persistent quality issues in occurrence data can impact the accuracy of scientific findings, underscoring the importance of filtering erroneous occurrence records in biodiversity analyses.

Innovation: We introduce an R package, occTest, that synthesizes a growing open-source ecosystem of biodiversity cleaning workflows to prepare occurrence data for different modelling applications. It offers a structured set of algorithms to identify potential problems with species occurrence records by employing a hierarchical organization of multiple tests. The workflow has a hierarchical structure organized in test*Phases* (i.e. cleaning vs. testing) *that encompass different testBlocks* grouping different *testTypes* (e.g. *environmental outlier detection*), which may use different *testMethods* (e.g. *Rosner test*, *jacknife*,etc.). Four different *testBlocks* characterize potential problems in geographic, environmental, human influence and temporal dimensions. Filtering and plotting functions are incorporated to facilitate the interpretation of tests. We provide examples with different data sources, with default and user-defined parameters. Compared to other available tools and workflows, occTest offers a comprehensive suite of integrated tests, and allows multiple methods associated with each test to explore consensus among data cleaning methods. It uniquely incorporates both coordinate accuracy analysis and environmental analysis of occurrence records. Furthermore, it provides a hierarchical structure to incorporate future tests yet to be developed.

Main conclusions: occTest will help users understand the quality and quantity of data available before the start of data analysis, while also enabling users to filter data using either predefined rules or custom-built rules. As a result, occTest can better assess each record's appropriateness for its intended application.

KEYWORDS
data cleaning, outlier, quality, R, species occurrence

1 | INTRODUCTION—SPECIES OCCURRENCE DATA QUALITY

Species occurrence data are a foundation for comparative biology and are a major source of information in ecology, evolution, biogeography and ultimately influence biodiversity preservation and conservation efforts (Lavoie, 2013; Shaffer et al., 1998). Such information provides important insights into species' ecological requirements (e.g. niches) (Franklin, 2010; Peterson et al., 2011), is the basis of range delineation (Gaston, 2003), can be used to estimate global and local abundances (Enquist et al., 2019; Gotelli et al., 2023), and can critically underlie conservation assessment, prioritization, monitoring (Boonman et al., 2024; Darrah et al., 2017; Enquist et al., 2019; Feng et al., 2021; Guo et al., 2022; Pelletier et al., 2018; Zizka et al., 2021), and climate change exposure analysis (Dawson et al., 2011; Serra-Diaz et al., 2014). The fidelity of species occurrence records is thus pivotal in developing a more predictive biodiversity science.

The tremendous efforts of biodiversity institutions worldwide in the last 20 years have boosted natural history collection digitization and mobilization of species occurrence data (Graham et al., 2004; Guralnick et al., 2007). In parallel, there has been a rise of citizen science programs and the development of mobile applications to record biodiversity for different regions and target different groups (iNaturalist <https://www.inaturalist.org/>, iMapinvasives <https://www.imapinvasives.org/>, etc.). These massive data inputs have led to biodiversity aggregators that differ in terms of taxonomic or geographic scope and scientific network but also with a high degree of redundancy (e.g. the same record repeated in different databases) among other data aggregators (Feng et al., 2022).

Despite significant advances in mobilizing occurrence data, biases persist within many biodiversity datasets (Enquist et al., 2016; Franklin et al., 2017). These biases undermine the representativeness and reliability of big biodiversity databases. Taxonomic, temporal, and spatial biases are still widespread (Maitner et al., 2023; Meyer et al., 2016). While continued acquisition of new data is essential for overcoming these limitations, it is equally crucial to refine the existing data through comprehensive analysis and integration of diverse data sources (Feng et al., 2022). A possible solution is to integrate different data sources and to analyse the quality of the current available data at hand.

Integrating datasets or databases is challenging, and it often leads to inconsistencies stemming from the underlying data due to reuse for different applications than intended, syntactical errors (typos), different data sources when checking errors, and irregularly incomplete metadata. Integration also comes with challenges, such as different taxonomic inconsistencies among different databases for which specific software for taxonomic standardization has been developed (Boyle et al., 2013; Grenié et al., 2023). Issues about data quality in species occurrence data have largely been identified (Engemann et al., 2015; Enquist et al., 2016, 2019; Feng et al., 2022; Franklin et al., 2017; Meyer et al., 2016; Serra-Diaz et al., 2017; Sillero et al., 2023). These include typical geographical errors such as swapping longitude and latitude, coordinates specifying museum location rather than specimen observation, or coordinates at 0–0 longitude-latitude when coordinates are not available (Maldonado et al., 2015), and may affect conservation prioritization and niche analysis if they remain undetected (Panter et al., 2020; Ribeiro et al., 2022). Compared to spatial, temporal and taxonomic gaps, these errors may only represent a fraction of the issues in biodiversity data. However, identifying them is crucial to increase the quality of the currently available data used in many analyses.

Biodiversity data infrastructures and data aggregators have implemented potential error flagging workflows to improve data quality. For instance, the Botanical Information and Ecology Network (BIEN, https://bien.nceas.ucsb.edu/bien/) performs a comprehensive set of tests assessing taxonomic issues, identifying potential non-native records, and flagging cultivated specimens among others (Enquist et al., 2019; Maitner et al., 2018). As another example, the Global Biodiversity Infrastructure Facility (GBIF; www.gbif.org) interprets original data from publishers and provides users with information on potential error issues including taxonomic issues, geospatial issues and recording issues (countries, time, elevation; https://data-blog.gbif.org/post/issues-and-flags/blog.gbif.org/post/issues-and-flags/). Besides these initial quality assessments, users need to determine the impact of such issues for a specific study objective. They are bound to perform a study-specific filtering of records.

Removing (e.g. *cleaning*, *scrubbing*, *and filtering*) low-quality or erroneous occurrences may delete a large proportion of the initial dataset, is time-consuming, and may have strong implications for the analysis. Despite warnings, careful analysis of data quality issues and their reporting are not widespread among the scientific community (Ball-Damerow et al., 2019). Current approaches to geographic data cleaning using R include packages for geographic error identification such as *coordinateCleaner* (Zizka et al., 2019) or occOutliers (Merow, 2023), packages that additionally implement some error corrections such as biogeo (Robertson et al., 2016), or custom-built workflows incorporating such tests (Jin & Yang, 2020; Ribeiro et al., 2022; Serra-Diaz et al., 2017). Further, packages that implement taxonomic information standardization and geographic cleaning methods have been released (Chamberlain, 2016; Gueta et al., 2018), and data standardization packages have appeared to deal with specific issues such as invasive species (Seebens et al., 2020). Tools for visually assessing potential bias in temporal or environmental trends are also available (Boyd et al., 2021). Ultimately, the results of this important data-cleaning step may thus vary depending on the tests that have been implemented, which may not be the same depending on the specific tests and methods used.

Tests for identifying potentially erroneous records are not failproof. Tests rely on assumptions, thresholds, or auxiliary data sources. For example, different statistical methods can be used to detect spatial outliers or different data sources may be used to detect urban areas or highly anthropogenic conditions that may signify human introductions or cultivated records. Performing multiple tests for the same type of issue (e.g. coordinate precision, outlier detection, etc.) can draw from different methods' strengths, enable an ensemble approach among tests and detect potential false positives.

Here, we present the R package occTest. In occTest we apply multiple tests for the same type of issues synthesizing available functions and adding newly developed environmental and accuracy tests. We further provide functions to generate summaries and plot results to guide users in selecting species occurrences.

The package is publicly available on gitHub (https://github.com/pepbioalerts/occTest.git), rather than CRAN, due to the large environmental datasets needed for use and examples.

2 | AN INTEGRATED APPROACH TO OCCURRENCE DATA QUALITY CONTROL AND ASSESSMENT

In occTest, we flag species occurrence records based on an organized hierarchy and scoring multiple tests for each dimension of species occurrence data (environment, geography, time, accuracy), building on the legacy of cleaning workflows such as BIEN and data issue identification from GBIF. Two novel aspects of this package are the use of multiple tests to analyse occurrence records with respect to a user-defined environmental (typically climatic) grid. Thus, we include tests to identify environmental outliers and multiple spatial and environmental accuracy issues related to the environmental grid of interest for the given study. This is similar to other approaches to assess environmental bias in records using graphs (occAssess, Boyd et al., 2021), but it differs as it implements multiple tests and numerically identifies the outliers, and the consequences of spatial (in)accuracies in outlier detection. The workflow is taxon agnostic, modular, and it can summarize results consistently among different issues encountered in occurrence records.

Tests are organized hierarchically into *phases*, *blocks*, *types*, and *methods* (Figure 1). *Phases* correspond to the major steps in the workflow: clean (remove blatant errors), test (apply different types of tests to characterize the remaining occurrences), and filter (remove records based on rules). Test *blocks* group tests that analyse a given dimension of the occurrence records. Four blocks are currently implemented: the geographic dimension (geo), the environmental dimension (env), *land use* and anthropogenic dimension (lu), and the temporal dimension (*time*). Test *Types* group different test *methods* to detect the same issue using various techniques. For instance, the test type *centroidDetection* implements two methods based on different packages (BIEN and coordinateCleaner) to identify whether occurrence records belong to a centroid of an administrative unit. For each test *type* (*testType)* and test block (*testBlock*), the output of occTest will give a score corresponding to the proportion of methods that flagged the record as potentially problematic (Figure 1). This scoring system enables providing the basis for an ensemble approach (e.g. consensus among methods) toward filtering occurrence records.

occTest is specifically designed to work together with the environmental data in a user's study. In occTest, we incorporate environmental analyses such as outlier detection in the environmental space and the impact of coordinate accuracy on species niche estimation. Thus, it is fit for species distribution modelling and niche analysis using biogeographical data. This will be a key feature given the increasing availability of higher spatial resolution environmental data, especially from remote sensing.

3 | occTest WORKFLOW OVERVIEW

The primary function of the package is *occTest()*. *occTest()* first standardizes input occurrence data and fields and performs initial checks on suitable input formats. The minimum set of information needed is a species name [character], a dataset with coordinates [data.frame], and spatial grids of environmental (e.g. climate) variables [raster]. Other information can be specified in the input data.frame or *occTest()* and will determine the scope of tests that can be performed. For example, the user can specify the countries where species are considered native versus invasive. Most of the input information (table columns) borrows from GBIF table structure, but users can

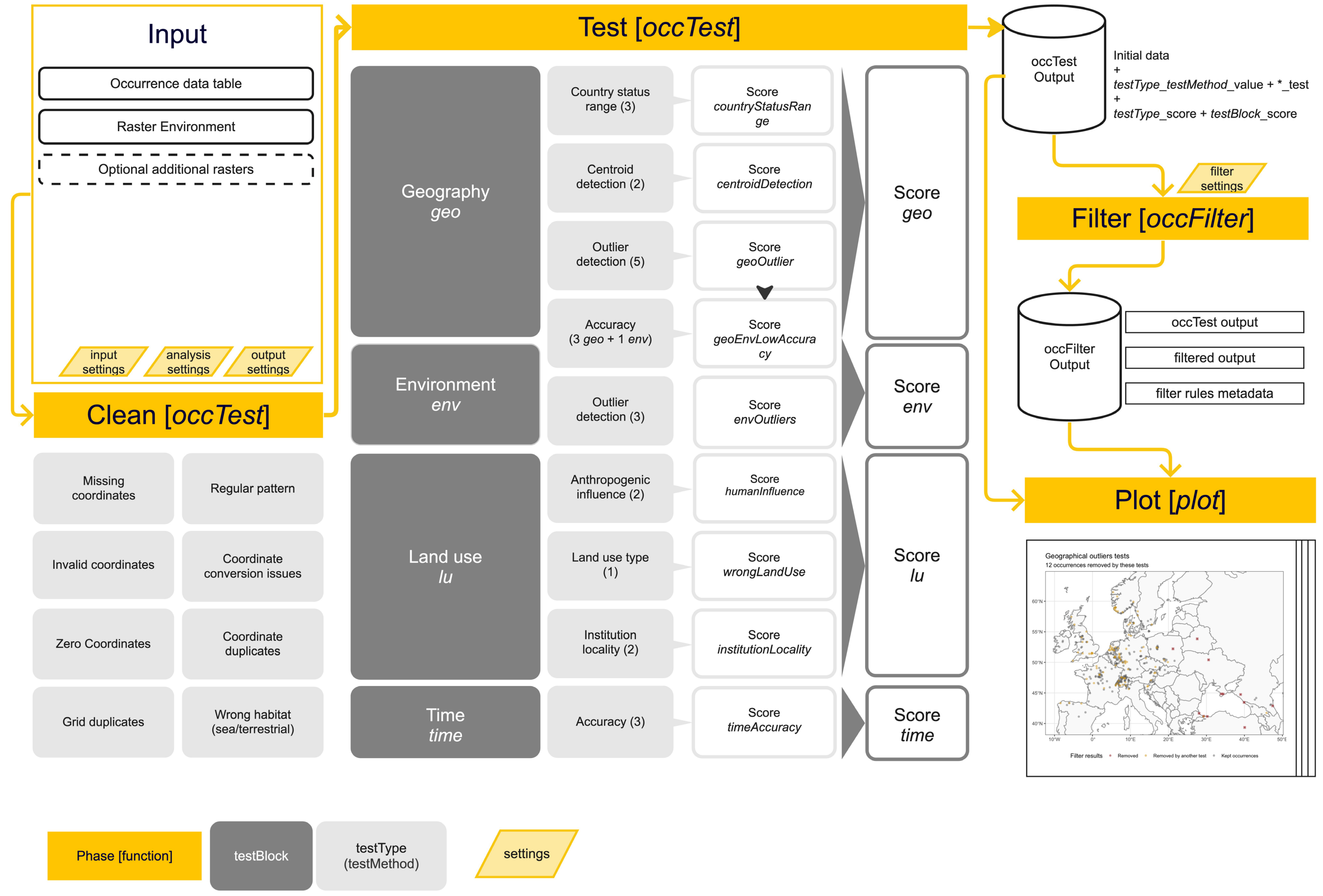


FIGURE 1 Diagram of the occTest workflow. The workflow requires input data and settings (light yellow box) to initialize the *occTest()* function, which will implement a series of quality tests in phases (yellow boxes) test blocks (dark grey boxes) and test types (light grey boxes). The first phase (Clean) involves applying several test types (*testTypes*) to filter out erroneous records. In the second phase (Test) a series of tests will characterize potential types of errors (e.g. centroids, outliers—light grey boxes) using several methods that can be used to identify such errors (numbers in parentheses see Table 1). Results are summarized in scores by *testType* or by *testBlock*. Test and scoring results are joined to the initial dataset to produce an *occTest* data.frame-type object. This object can be filtered (*occFilter()*) according to several either pre- or user-defined rules. occTest and occFilter outputs can be plotted using plot-generic methods.

TABLE 1 Tests performed during the cleaning phase (Figure 1). It integrates functions *coordinateCleaner* (Zizka et al., 2019) and biogeo (Robertson et al., 2016).

testTypes	testMethods	Flagged records	Package(s)
coordIssues	missing	Missing coordinates	occTest
	invalCoord	Invalid coordinate format	Modified from biogeo
	zeroCoord	lat = 0, long = 0 coordinates	coordinateCleaner
	coordConv	Pattern interval	Modified from coordinateCleaner in occTest
duplicates	dupGrid	Records in the same gridcell in the environmental value	occTest
	dupExact	Records with the exact same coordinates	Modified from biogeo

set custom column names directly (see example 2 below). Notably, some tests may require large data sets that need to be downloaded (e.g. raster of human influence index), and occTest provides defaults for automated downloads.

In the first cleaning phase, the workflow identifies typical known errors like missing coordinates, coordinates at latlong (0,0), and in wrong environments (e.g. land/sea). It further applies a correction method for land/sea reassignment (Robertson et al., 2016). That is, for a terrestrial

TABLE 2 Tests performed during the *test* phase (Figure 1). It integrates functions or modified versions of them from packages *coordinateCleaner* (Zizka et al., 2019), occOutliers (Merow, 2023), biogeo (Robertson et al., 2016), flexsdm (Velazco et al., 2022) and functions used in the BIEN workflow (Maitner et al., 2018).

testTypes	testMethods	Flagged records	Package
countryStatusRange	wrongNTV	Records located outside the native range (country-level)	occTest
	wrongCTRY	Incongruence between coordinate location and country of record	occTest, coordinateCleaner
	wrongINV	Records located outside the alien range (country-level)	occTest
centroidDetection	BIEN	Records located in the centroid of several administrative entities or nearby	BIEN*
	CoordinateCleaner	Records located in the centroid of several administrative entities	coordinateCleaner* (cc_cap, cc_cen)
geoOutliers	alphaHull	Records outside an alphaHull shape (default 2)	occTest
	distance	Records whose distance to other records is >*x*, *x* = 1000 by default, can be changed	Modified from coordinateCleaner* cc_outl(method='distan ce')
	median	Their mean distance to all other records of the same species is larger than mltpl * the interquartile range of the absolute median distance of all records of this species	Modified from coordinateCleaner* cc_outl (method='mad')
	quantileSamplingCorr	Their mean distance to all other records of the same species is larger than mltpl * the interquartile range of the mean distance of all records of this species	Modified from coordinateCleaner * cc_outl (method='quantile')
	Grubbs	A statistical test to determine if a distribution has either one or two outliers. The Euclidean distance between each occurrence record and the centroid of all records, in either geographic or environmental space, is calculated, and the Grubbs outlier test is performed on this distribution	Modified from occOutliers*
	Dixon	Performs Dixon test to identify outliers.	Modified from occOutliers*
	Rosner	Perform Rosner's generalized extreme Studentized deviate test for up to k potential outliers in a dataset	
	Mcp	Records that are excluded from the minimum convex polygon that was built using the selected threshold (default to 95%) as percentage of the records of this species	occTest
geoenvLowAccuracy	lattice	Occurrences originating from a rasterized/lattice scheme	CoordinateCleaner* (cc_round)
	elevDiff	Difference between recorded elevation and raster elevation >threshold (default 100 m)	occTest
	percDiffCell	Percentage of times the record falls in a different cell grid of the environmental raster given an accuracy measure	occTest
humanDetection	HumanInfluence	Records with high human modification (Kennedy et al., 2019), but user may provide preferred data source)	occTest
	UrbanAreas	Records in urban areas	coordinateCleaner
landUseType	wrongLU	Records in a specified land use (user provides data source)	occTest
institutionLocality	fromBotanicLocalityName	Records likely to be in a botanical garden from locality	occTest
	fromCoordinates	Records located in biodiversity institutions or GBIF headquarters	coordinateCleaner (cc_inst and cc_gbif)

(Continues)

TABLE 2 (Continued)

testTypes	testMethods	Flagged records	Package
geoenvLowAccuracy	envDiff		occTest
envOutliers	missingEnv	Missing environmental information	occTest
	bxp	Records 1.5 times beyond the length of the box in the boxplot	occTest
	Grubbs	Records with euclidean environmental distance far from the centroid of all records, based on a Grubbs outlier test with with *p* value <0.005	Modified from occOutliers
	Dixon	Performs Dixon test to identify outliers.	Modified from occOutliers
	Rosner	Perform Rosner's generalized extreme Studentized deviate test for up to k potential outliers in a dataset	Modified from occOutliers
	jack	Reverse jackknife method detection of environmental outliers	flexsdm
	svm	Supported vector machine method for detection of environmental outliers. Absence data simulated based on a buffer zone around presences corresponding to the median distance among presence points. Absences correspond to 60% of the area and less than 10,000 absence points	flexsdm
	rf	Random forests method for detection of environmental outliers. Absence data simulated based on a buffer zone around presences corresponding to the median distance among presence points. Absences correspond to 60% of the area and less than 10,000 absence points	Modified from flexsdm
	rfout	Random forests outlier method for detection of environmental outliers. Absence data simulated based on a buffer zone around presences corresponding to the median distance among presence points. Absences correspond to 60% of the area and less than 10,000 absence points	Modified from flexsdm
	lof	Local outlier factor method for detection of environmental outliers	flexsdm
timeLowAccuracy	noDate	No timestamp available for the record	occTest
	noDateFormatKnown	Timestamp format not known	occTest
	outDateRange	Timestamp out of a range of dates	occTest

species (default option, otherwise habitat='sea' in *occTest()*) with an occurrence record in the sea but adjacent to the coastline, the record will be relocated to the nearest grid cell in a terrestrial environment—and keep the original coordinates as output together with modified ones. *occTest()* differentiates true vs. grid-dependent duplicates (Table 1); that is whether the coordinates of two records match exactly or simply fall within the same grid cell of the environmental data (Figure 1). This information may help develop sampling effort or sampling bias layers. After these initial filtering tests, the workflow sets aside records that did not pass these cleaning *phase* tests. This initial filtering speeds up the subsequent tests and increases the reliability of some tests (e.g. outlier detection).

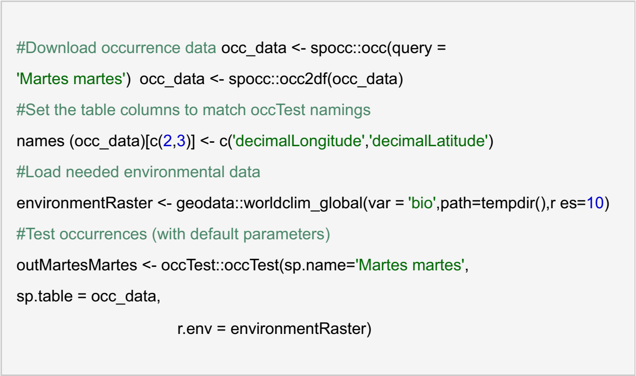
A series of test types (testTypes) organized in larger blocks (testBlocks) are then applied to the remaining occurrences during the *test* phase (Figure 1 and Table 2). First, the workflow identifies occurrences from regions where the species is native or non-native—should the user provide such information—and tests the match between coordinates and the country where the record has been reported. Subsequently, several testMethods are applied to identify issues in geographical space, and land use space (checks related to anthropogenic issues like location in biodiversity institutions or in urban areas), environmental space (occurrences in suspicious marginal habitats or without associated environmental information), temporal space (issues related to dates of the record). In addition, we developed new testMethods and testTypes to quantify how coordinate accuracy may affect the uncertainty around a climate variable at the location (Table 2). These are relevant when working at higher resolution climate variables.

The workflow output is the initial occurrence data.frame with additional columns indicating the results of the tests. The additional columns indicate the results of each test following a structured nomenclature of *<testType>_<testMethod>_value* and *<testType>_<testMethod>_test* (Figure 1). The *_value column indicates the result of the test (numeric or logical). In contrast, the *_test is a logical output (True or False) indicating if the record is flagged according to that specific test and method. In some tests, there is also a column named *_comments where details of the methods are specified (see Example 1). For instance, it will show the threshold of >45 used to detect high human influence (default to (Kennedy et al., 2019), but may be provided by the user) in a record. NA values indicate that a test was not performed.

4 | EXAMPLE 1: SIMPLE TESTING

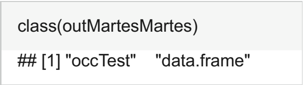
The most minimal input consists of an input data.frame (with key specified data columns) and a multilayer raster object with environmental data. The occTest function default values follow GBIF naming conventions, so you must reformat the table to match those names or modify the settings (see Example 2).

As testing can be slow, particularly with very large datasets, occTest provides status messages that report the time elapsed in each test. It can be controlled with the parameter verbose in *occTest()*.



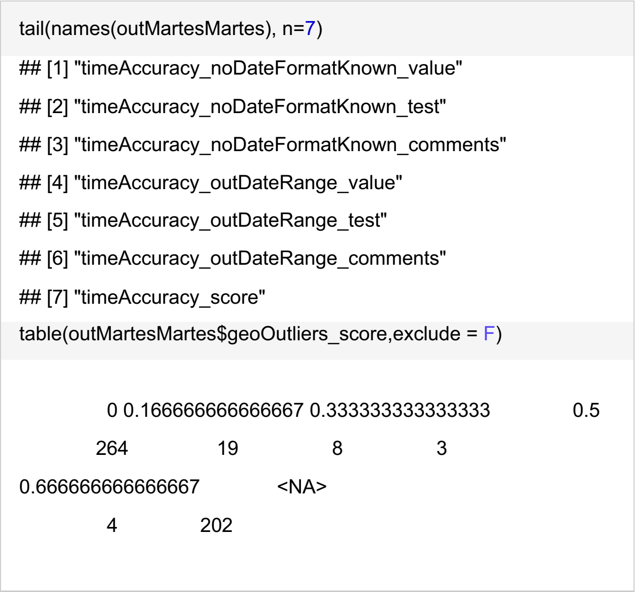
```
#Download occurrence data occ_data <- spocc::occ(query =
'Martes martes')  occ_data <- spocc::occ2df(occ_data)
#Set the table columns to match occTest namings
names (occ_data)[c(2,3)] <- c('decimalLongitude','decimalLatitude')
#Load needed environmental data
environmentRaster <- geodata::worldclim_global(var = 'bio',path=tempdir(),r es=10)
#Test occurrences (with default parameters)
outMartesMartes <- occTest::occTest(sp.name='Martes martes',
sp.table = occ_data,
                          r.env = environmentRaster)
```

The result is a data.frame of class occTest, which adds to the initial data.frame a series of columns specifying the tests.



```
class(outMartesMartes)
## [1] "occTest"    "data.frame"
```

We can identify and select a given test through column nomenclature of <testBlock>_<testType>_test, or identify scores (proportion of failed tests) in <testBlock>_score. For instance, one can easily identify in our dataset the number of occurrences that failed to pass a given rate of geographical outlier tests as:



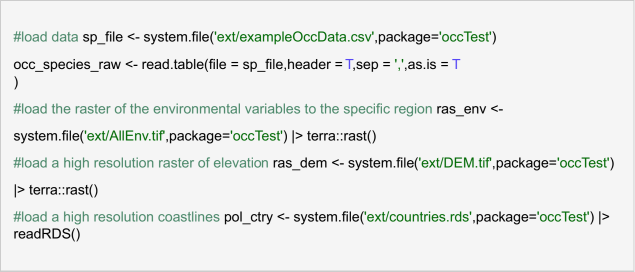
```
tail(names(outMartesMartes), n=7)
## [1] "timeAccuracy_noDateFormatKnown_value"
## [2] "timeAccuracy_noDateFormatKnown_test"
## [3] "timeAccuracy_noDateFormatKnown_comments"
## [4] "timeAccuracy_outDateRange_value"
## [5] "timeAccuracy_outDateRange_test"
## [6] "timeAccuracy_outDateRange_comments"
## [7] "timeAccuracy_score"
table(outMartesMartes$geoOutliers_score,exclude = F)


         0 0.166666666666667 0.333333333333333          0.5
       264              19               8              3
0.666666666666667          <NA>
         4           202
```

In this case, three occurrences had 50 percent of the tests flagging them as outliers in the geographic space. Note that the 212 records where the score is NA represent occurrences where the tests have not been performed. Typically, this is because these records have been removed by the initial cleaning phase of erroneous records (e.g. occurrence in the sea, etc.). All output summary columns are available in Appendix S1.

5 | EXAMPLE 2: USER-DEFINED INPUTS

We use a predefined example dataset stored in the occTest package to showcase customization of the default parameters. It consists of a virtual species with occurrence records spread in the Mediterranean basin between France and Spain. The species is considered native in the Iberian Peninsula and invasive in France (Figure 2). In this example, we input raster for elevation and climate to a specific vector file with a coastline that is more detailed than the default layer of the occTest package. We also want to isolate occurrences in highly human influenced regions.



```
#load data sp_file <- system.file('ext/exampleOccData.csv',package='occTest')
occ_species_raw <- read.table(file = sp_file,header = T,sep = ',',as.is = T
)
#load the raster of the environmental variables to the specific region ras_env <-
system.file('ext/AllEnv.tif',package='occTest') |> terra::rast()
#load a high resolution raster of elevation ras_dem <- system.file('ext/DEM.tif',package='occTest')
|> terra::rast()
#load a high resolution coastlines pol_ctry <- system.file('ext/countries.rds',package='occTest') |>
readRDS()
```

The user can define which tests to turn off or modify through three parameters in the occTest function: *tableSettings*,

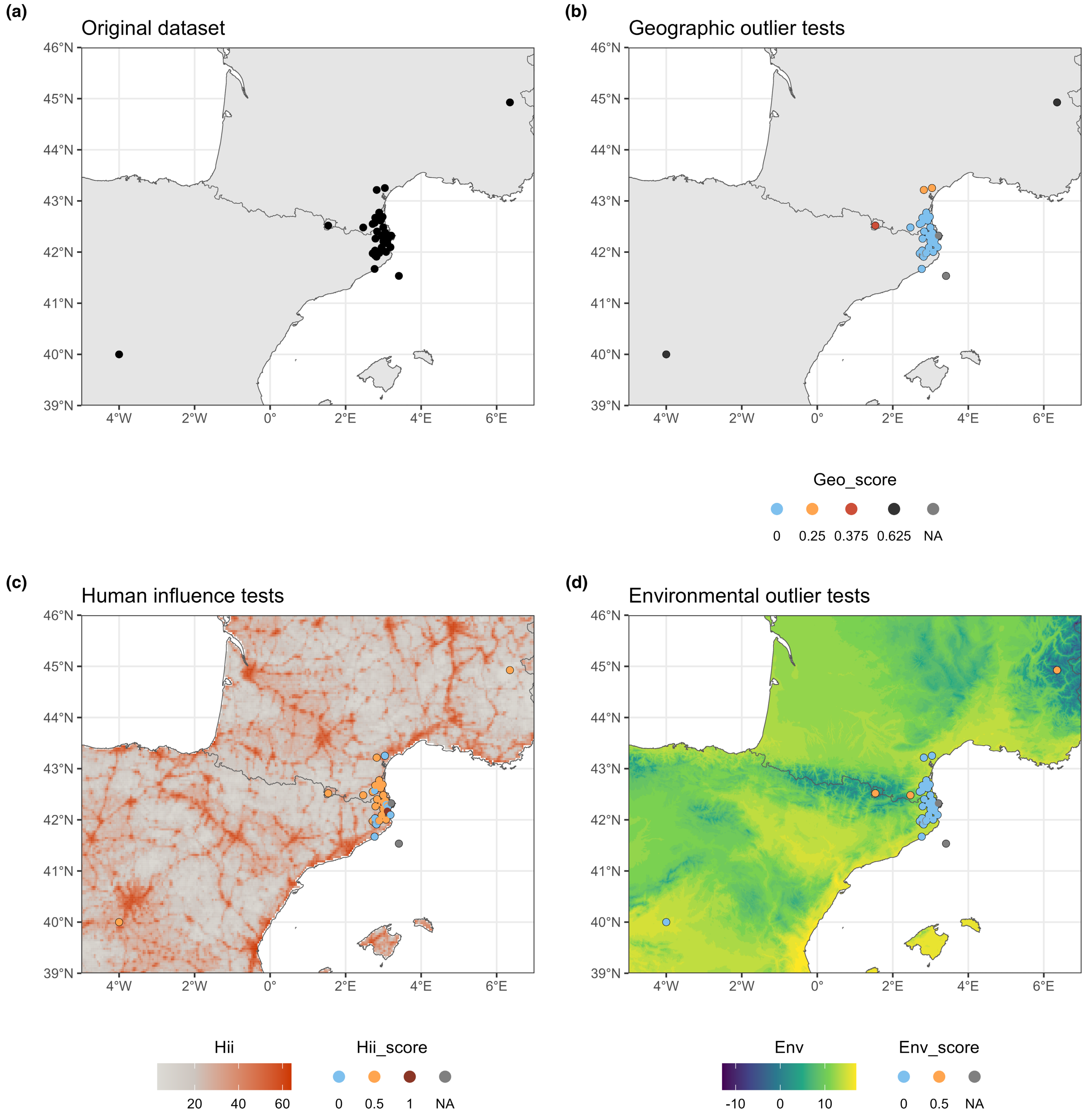


FIGURE 2 Plot showing a virtual species (a) passing through the geographical outlier tests (b), the human influence tests (c) and the environmental outlier tests (d). The extent of the plotting area has been modified to match the input raster data for display purposes, as some occurrences are far from the range to simulate different types of outliers.

analysisSettings and writeoutSettings. These consist of three named lists specifying (1) the input information (*tableSettings*), (2) the type and naming of outputs (*writeoutSettings*), and (3) the types of tests implemented and their parameters (*analysisSettings*). These three parameters are set to NULL by default in *occTest()* and internally loaded via the *defaultSettings()* function. However, these can be modified via the set_* family of functions, which provide shortcuts to load and modify default functions.

Input occurrence column names can be specified via the *set_tableNames()* (e.g. changing defaults). One can turn on or off certain *testTypes* or *testBlocks* through *set_testTypes()* or *set_testBlocks()*, or to specify outputs, *set_writeout()*. Here, we provide a quick example where we change column names read by *occTest()* by default (rather than changing the names in the initial occurrence data table to match the defaults of *occTest*). We additionally set the geoEnvAccuracy tests off and ask the workflow to write all outputs and tables derived to disk.

```
#show default settings
show_tableSettings() #modify
settings
                    my_table_Settings <- set_tableNames(x.field = 'MAPX'
                                                        y.field = 'MAPY',
                          t.field = 'DATE',
                          l.field = 'LOCALITYNAME',
                          c.field = 'CONTRYRECORD',
                          e.field = 'ELEVATION',
                          a.field =c('UNCERTAINTY_X_M','UNCERTAIN TY_Y_M'))
my_analysis_settings <- set_testTypes(geoenvLowAccuracy = T)
my_writeout_settings <- set_writeout(output.dir = getwd(),writeAllOutput =
T)
#run workflow
occ_species_test <- occTest(sp.name = 'Genus_species',
          sp.table = occ_species_raw,
          r.env =ras_env,
          tableSettings = my_table_Settings,                    analysisSettings =
          my_analysis_settings,                    writeoutSettings =
          my_writeout_settings,
          r.dem =ras_dem,
          ntv.ctry = 'ESP',
          inv.ctry = 'FRA',                    verbose = T)
```

```
#load default structure of settings default_settings <- defaultSettings()
my_analysis_settings <- default_settings$analysisSettings
#modify some analysis table parameters
my_analysis_settings$humanAnalysis$th.human.influence <-50
my_analysis_settings$rangeAnalysis$countries.shapefile <-pol_ctry
my_analysis_settings$rangeAnalysis$countryfield.shapefile <- 'ISO'
my_analysis_settings$geoenvLowAccuracy$doGeoEnvAccuracy <- F
my_analysis_settings$envOutliers$methodEnvOutliers <- "grubbs"


#run occTest with my new parameters occ_species_test occTest(sp.name =
'Genus_species',
          sp.table = occ_species_raw,
          r.env =ras_env,
          tableSettings = my_table_Settings,
          analysisSettings = my_analysis_settings,                writeoutSettings =
          my_writeout_settings,
          r.dem = ras_dem,
          ntv.ctry = 'ESP',
          inv.ctry = 'FRA',
          verbose = T)
```

To customize internal parameters and default data, we recommend using *defaultSettings()* to load the default structure of the occTest parameter settings *tableSettings*, *analysisSettings* and *writeoutSettings*, and modify internal values based on that structure. A complete analysis is performed with *fullSettings*(). A full guide to settings is provided in show_* family of functions: *show_tableSettings()*, *show_writeoutSettings()*, and *show_analysisSettings()*.

As an example of how to do so, we provide code where we (i) modify the threshold to 50 for the human influence index to eliminate records under strong human influence, (ii) use highresolution countries polygons, (iii) turn off the whole set of test types for the geoenvironmental accuracy analysis, and (iv) only use the 'Grubbs' method (Table 1) to identify outliers in the environmental space. This example further illustrates the structure of the lists in occcTest settings parameters. For better display of the list, we recommend using the *listviewer* package (de Jong et al., 2023). The results of the filtering approach can be seen in Figure 2.

6 | EXAMPLE 3: FILTERING occTest OUTPUT

occFilter() automatically filters results from *occTest()*. Two parameters control how occurrences are removed: (1) the error acceptance threshold (*errorThreshold*) and (2) the level at which the threshold is applied (by, e.g. *testTypes* or *testBlocks*). *errorThreshold* is a numeric value from 0 to 1 removing records above a certain specified level, which can be at the testType level or the testBlock level. For instance, when errorThreshold = 0.5 and by="testTypes", the function will filter records in which more than 50% of test types fail (e.g. geographic outlier test, environmental outlier tests, etc., Figure 1). These thresholds are not applied to the tests in the initial cleaning phase because these have already failed the basic requirements for inclusion in most analyses (Figure 1). The user can see classification of testTypes and testBlocks using *show_tests()*.

```
occ_species_test <- occTest(sp.name = 'Genus_species',
          sp.table = occ_species_raw,
          r.env =ras_env,
          tableSettings = my_table_Settings,
          r.dem =   ras_dem,
          ntv.ctry = 'ESP',
          inv.ctry = 'FRA',
          verbose = T)
outFiltered_byTestType  <- occFilter(df = occ_species_test,
          errorThreshold = 0.8,
          by = 'testType')
str(outFiltered_byTestType,max.level = 1)
```

```
## List of 3
##  $ filteredDataset: tibble [6 × 146] (S3: tbl_df/tbl/data.frame)
##   ..- attr(*, "Settings")=List of 3
##  $ summaryStats   :'data.frame': 50 obs. of  27 variables:
##  $ rule           :'data.frame': 8 obs. of  2 variables:
##  - attr(*, "class")= chr [1:2] "occFilter" "list"
##  - attr(*, "Settings")=List of 3
```

The output of occFilter is a list of class “occFilter” with three named elements: (1) filteredDataset—the filtered initial dataset, (2) summaryStats—showing the number of tests performed, the overall score per testType, and (3) rule—a data.frame with the thresholds used per test.

The user can set test-specific thresholds via the *custom* parameter in *occFilter()*. For guidance, we provide some arbitrarily defined thresholds that can be used through the parameter *errorAcceptance* and corresponding to filtering philosophies of a “relaxed” user (*errorThreshold* = 0.7), a “strict” user (*errorThreshold* = 0.2) or a diplomatic user who will filter according to a “majority” rule (*errorThreshold* = 0.5). We provide here a code example to filter the above dataset at the testType level under a ‘relaxed’ philosophy.

Summary columns are provided and identified as <*testType*>_*score*. This score corresponds to the fraction (0–1) of methods that flagged a given record for a given test type. For example, a score of 0.5 in the *envOutliers_score* column indicates that half of the tests related to detecting outliers in the environmental space flagged the occurrence as problematic (see Example 1).

The overall structure and naming conventions allow for additional methods and test types to be implemented. It also facilitates a transparent output where users can remove records based on their specific case study. We additionally provide *occFilter()* that automatically removes occurrences based on default thresholds on each test type score (Example 3). Nevertheless, we strongly encourage users to customize their filtering to ensure that the quality assessment is suitable for the study (Veiga et al., 2017).

7 | EXAMPLE 4: PLOTTING occTest

We provide a generic plot function for occTest. The method plot.occTest allows to plot the initial coordinates filter phase (Figure 1) when only the occTest output is provided (Figure 2a).

```
#getting tree occurrences from a central European oak tree
df_qupe <- spocc::occ (query ='Quercus petraea',limit = 1000) occ_data_qupe <-
spocc::occ2df(df_qupe)
names (occ_data_qupe)[c(2,3)] <- c('decimalLongitude','decimalLatitude')
#downloading environmental data
renv <- geodata::worldclim_global(var = 'bio',path=tempdir(),res=10)
#running occTest out_qupe <- occTest(sp.name='Quercus
petraea',
          sp.table = occ_data_qupe ,
          r.env = renv,
          verbose = F)
plot(out_qupe)
```

When the result of the filtering process (described in Example 3) is provided as a second argument in the plot function, a plot for each filtering result testType or testBlock is mapped (Figure 2b–d). If the user has chosen to perform a custom filtering with the ‘custom’ parameter of *occFilter()*, the plots can also be produced.

```
# running a strict filtering process
out_filtered_qupe <- occFilter(df = out_qupe , errorAcceptance = 'strict')
# adding the occFilter object to the plot
list_of_plots <- plot(x = out_qupe ,
occFilter_list = out_filtered_qupe, show_plot = F)
list_of_plots
```

The maps returned by plot.occTest are shown in Figure 3. When possible, a visual verification of the filtering process should be performed as it checks the relevance of tests and thresholds applied. This enables adjusting the *errorAcceptance*. In our *Quercus petraea* example, the land use and human influences tests might be too stringent and remove 59 records (Figure 3). Keep in mind that plotting many occurrences may hamper the correct display of the filtering process. The user can still use the output from occTest and occFilter to create custom displays and verifications of the test value and filtering process, respectively.

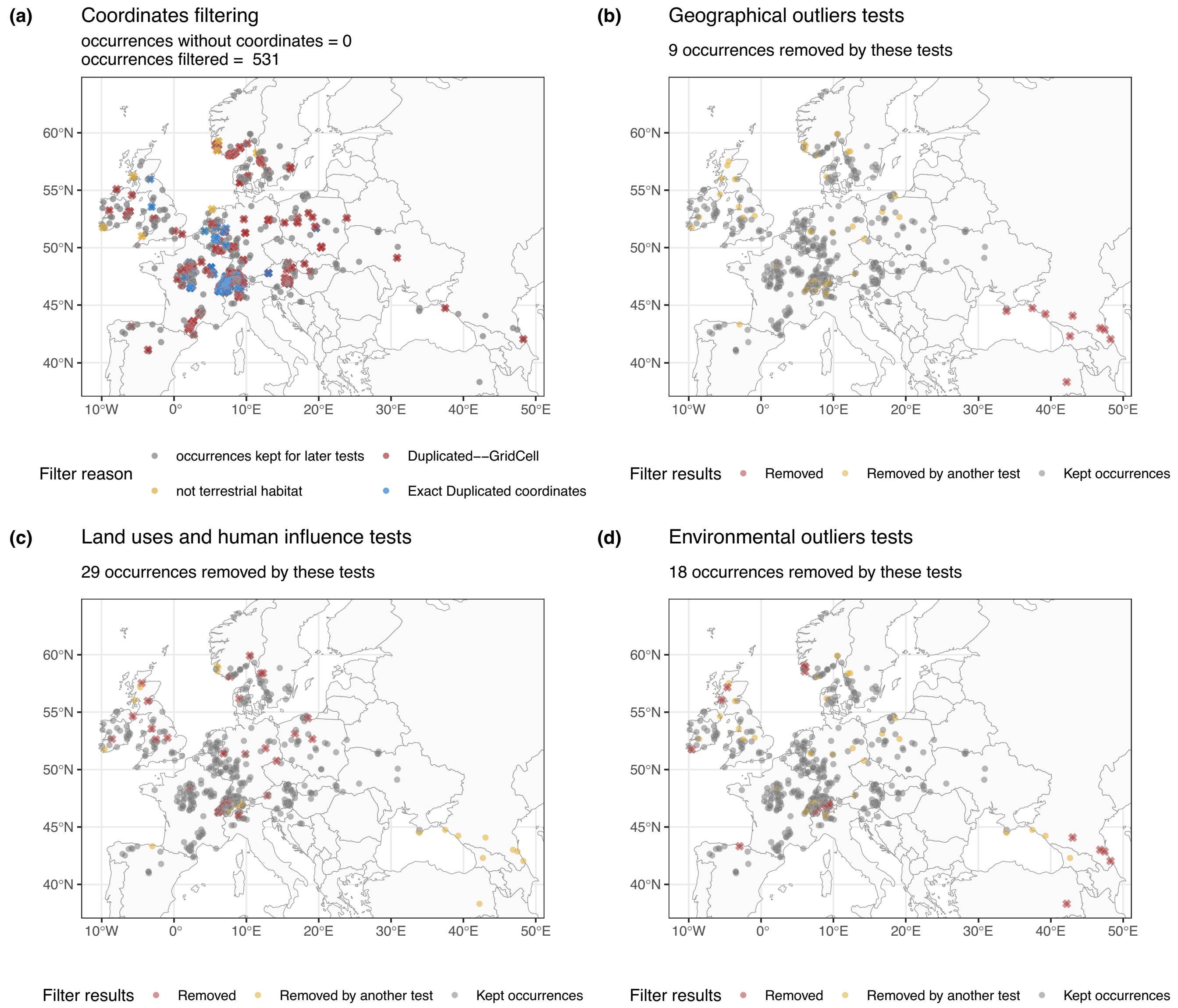

(a)
Coordinates filtering
occurrences without coordinates = 0
occurrences filtered = 531
60°N
55°N
50°N
45°N
40°N
10°W
0°
10°E
20°E
30°E
40°E
50°E
Filter reason
● occurrences kept for later tests
● Duplicated−−GridCell
● not terrestrial habitat
● Exact Duplicated coordinates
(b)
Geographical outliers tests
9 occurrences removed by these tests
Filter results
● Removed
● Removed by another test
● Kept occurrences
(c)
Land uses and human influence tests
29 occurrences removed by these tests
Filter results
● Removed
● Removed by another test
● Kept occurrences
(d)
Environmental outliers tests
18 occurrences removed by these tests
Filter results
● Removed
● Removed by another test
● Kept occurrences

FIGURE 3 Output of the generic plotting function for an *occTest* object when both the *occTest* output and the *occFilter* output are provided. Results shown for (a) the filtering phase, and for three *testBlocks*: (b) geographical outliers, (c) land and human influence tests, and (d) environmental tests. Map extent has been reduced, as it included some outlier occurrences in Australia, to improve the readability.

8 | EXAMPLE 5: ADDING YOUR OWN TESTS

The user can incorporate customized *testMethods* that may be important for a specific analysis, and subsequently recompute scores with *reTest()* and filter occurrences. The user must supply a data.frame with column names using the same nomenclature used in occTest: *testType_testMethod_test* (logical). We encourage the user to additionally supply the *testType_testMethod_value* (numeric or logical) and *testType_testMethod_comment* (character) but only the *testType_testMethod_test* is required. Here we continue with Example 4 data to incorporate a new (simulated) *testMethod* in the humanDetection testType:

```
myNewTest <- data.frame (humanDetection_myInventedTest_test=sample (c(T,F), replace = T,size =
nrow(out_qupe)))
out_test_newAdded <- reTest(occTest_result = out_qupe,
my_new_test =  myNew Test)
df_compare <- data.frame (new_score= out_test_newAdded$humanDetection_score
,
old_score = out_qupe$humanDetection_score)
head (df_compare)
##   new_score old_score
## 1 0.3333333       0.5
## 2 0.0000000        NA
## 3 0.6666667       0.5
## 4 0.3333333       0.5
## 5 0.6666667       0.5
## 6 0.3333333       0.5
```

9 | occTest PERFORMANCE AND TESTS AGREEMENT

We tested the package behaviour on a sample of 1000 species (Appendix S2). We randomly extracted a species list from the Global Tree Search List (v1.6, Beech et al., 2017) and queried the Global Biodiversity Infrastructure Facility (GBIF www.gbif.org [GBIF.Org User, 2023]). Because taxonomies do not perfectly match among the two databases, the final set of species in this test is 907. For each species, the number of occurrences ranged from 1 to 51,848 occurrences. We tested these occurrences with a standard climate data of mean annual temperature and annual precipitation at 30 arcsec resolution (worldclim v2, Fick & Hijmans, 2017). We ran *occTest()* with full settings, including tests of geo-environmental accuracy. Still, we did not perform land use-related tests because land-use type exclusion is a specific test that depends on the analysis. In contrast, we seek general information on workflow elapsed times.

The median processing time for *occTest()* is 18.34 s per species (11.62 s interquartile range) on an Apple M2 Max in R version 4.3.0. Time elapsed varied with sample size, although it peaked at ca. 50 s (Figure 4). The maximum elapsed time for a single species was recorded for *Ailanthus altissima (Mill.) Swingle*, with 51,848 records and 2690.93 s of testing time.

Across different tests implemented, we observe an overall agreement in tests of around 90% across methods for each testType (Figure 5a), except for the tests related to accuracy in geographic and environmental space. This large agreement is due to the many records that passed all the implemented tests. Upon restricting results to records flagged with potential issues by at least one testMethod, consensus among methods is noticeably lower. A minimal percentage of cases exist where all methods unanimously identify a record as a potential issue, as illustrated in Figure 5b. The low consensus rate is particularly noticeable for tests related to outlier detection in environmental and geographic spaces and those related to accuracy level. The little variation around some tests is due to only 2 tests being performed. Hence, 0.5 is the only possible value when testMethods disagree. This disagreement showcases the need to further evaluate different dimensions and methods to identify potential issues in occurrence data.

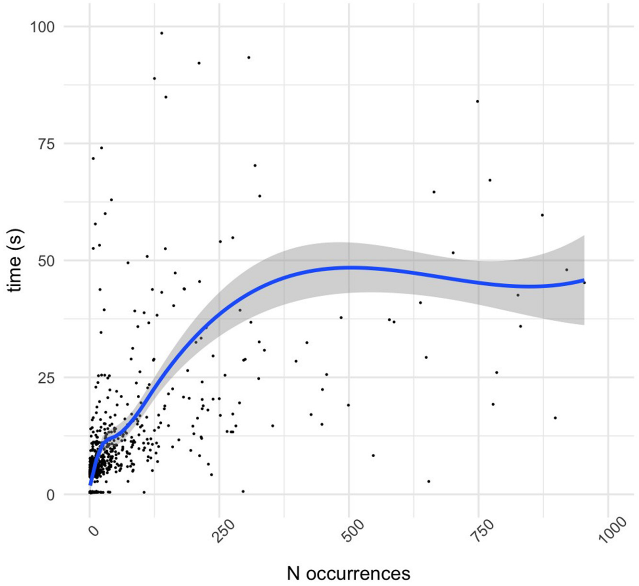


FIGURE 4 Time elapsed to run *occTest()*. Dots represent species and the blue line represents a smooth interpolation trend line using local polynomial regression fitting. *X* axis is truncated at 1000 occurrences for aesthetics.

10 | BEST PRACTICES FOR PACKAGE APPLICATION

occTest was built to apply and incorporate different tests in a single workflow with a common taxonomy of potential errors to facilitate data preparation for biogeographic analysis. The use of automatic workflows to detect errors, however, should be used with extreme caution. First, every taxon and region may hold different biases affecting quality tests. For instance, outlier detection is affected by different sampling strategies across species ranges. Human influence should be coupled with temporal analysis and may thus be a desired characteristic of the study. As a result, the human dimension tests applied in occTest may not be a suitable filtering options and should be turned off. Grid-duplicates are important to account for sampling effort and can be used in analyses as to understand biodiversity data completeness (Lobo et al., 2018). All in all, occTest is best used to identify and explore such potential errors, but identifying the source of such errors and correct for them should be performed by the researcher, based on the taxa knowledge, and aims of the study. We encourage the deep exploration of the tests before performing automated filtering approaches, and to use occTest to help report on species distribution models steps (see ODMAP; Zurell et al., 2020) and to assess the potential for risk of bias in occurrence data samples (see ROBBIT, Boyd et al., 2022).

11 | SUMMARY

The occTest package provides a synthetic way of applying common and new tests (spatial accuracy, environmental space outliers) for potential issues in species occurrence data. A vignette with examples is available together with a package on Github. An in-depth vignette can be found at vignetteXtra-occTest.

ACKNOWLEDGEMENTS

We thank the help and inspiration from Brad Boyle. This work is funded with the support of NASA Grant 80NSSC22K0883 (JMSD and CM), NSF 2225078 (CM) and ANR-21-CE32-0003 (JMSD). JB acknowledges the funding from the joint AgroParisTech and Région Grand-Est PhD grant 19_GE8_01020p05035. JCS considers this work a contribution to Center for Ecological Dynamics in a

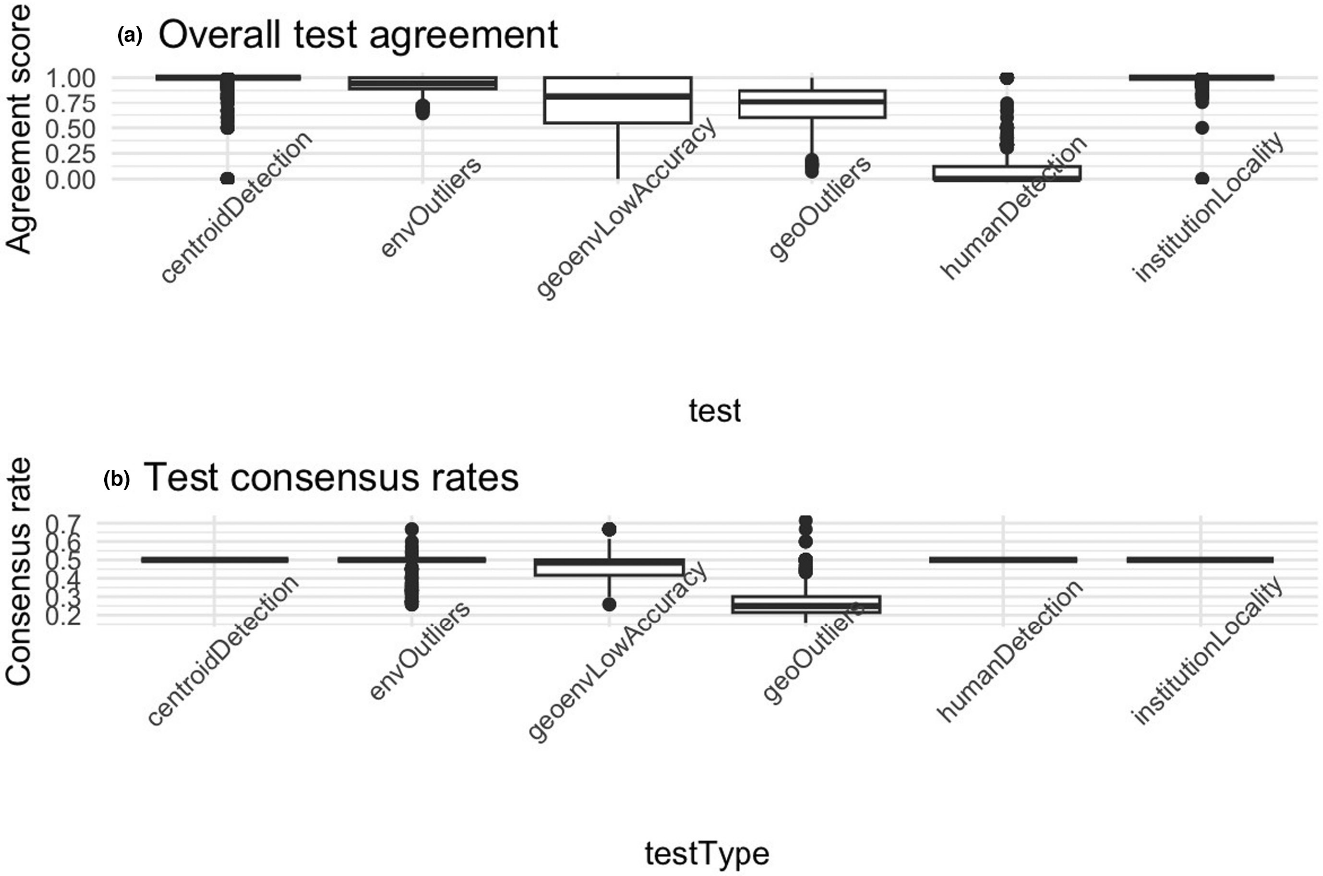


FIGURE 5 Agreement and consensus rates between *testMethods* implemented in different *testTypes* for every species: (a) agreement indicates the ratio of records per species where tests implemented agree, and it includes both records with and without identified issues, (b) indicates consensus levels—the ratios of agreement when at least one *testMethod* identified a potential issue for a given *testType*.

Novel Biosphere (ECONOVO), funded by Danish National Research Foundation (grant DNRF173) and his VILLUM Investigator project "Biodiversity Dynamics in a Changing World", funded by VILLUM FONDEN (grant 16549).

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The R package is available through github at. https://github.com/pepbioalerts/occTest.git. The data that supports the findings of this study are available in the supplementary material of this article. For long-term version storage the package will be available on Zenodo/DRYAD upon manuscript acceptance. Data sources to run the examples are public and can be found on GBIF.org (https://doi.org/10.15468/DL.H3J2WJ) and Worldclim (www.worldclim.org).

ORCID

Josep M. Serra-Diaz https://orcid.org/0000-0003-1988-1154
Jeremy Borderieux https://orcid.org/0000-0003-3993-1067
Coline C. F. Boonman https://orcid.org/0000-0003-2417-1579
Wen-Yong Guo https://orcid.org/0000-0002-4737-2042
Brian J. Enquist https://orcid.org/0000-0002-6124-7096
Jens-C. Svenning https://orcid.org/0000-0002-3415-0862
Cory Merow https://orcid.org/0000-0003-0561-053X

REFERENCES

Ball-Damerow, J. E., Brenskelle, L., Barve, N., Soltis, P. S., Sierwald, P., Bieler, R., LaFrance, R., Ariño, A. H., & Guralnick, R. P. (2019). Research applications of primary biodiversity databases in the digital age. *PLoS ONE*, *14*, e0215794.

Beech, E., Rivers, M., Oldfield, S., & Smith, P. P. (2017). GlobalTreeSearch: The first complete global database of tree species and country distributions. *Journal of Sustainable Forestry*, *36*, 454–489.

Boonman, C. C., Serra-Diaz, J. M., Hoeks, S., Guo, W.-Y., Enquist, B. J., Maitner, B., Malhi, Y., Merow, C., Buitenwerf, R., & Svenning, J.-C. (2024). More than 17,000 tree species are at risk from rapid global change. *Nature Communications*, *15*, 166.

Boyd, R. J., Powney, G. D., Burns, F., Danet, A., Duchenne, F., Grainger, M. J., Jarvis, S. G., Martin, G., Nilsen, E. B., Porcher, E., Stewart, G. B., Wilson, O. J., & Pescott, O. L. (2022). ROBITT: A tool for assessing the risk-of-bias in studies of temporal trends in ecology. *Methods in Ecology and Evolution*, *13*, 1497–1507.

Boyd, R. J., Powney, G. D., Carvell, C., & Pescott, O. L. (2021). occAssess: An R package for assessing potential biases in species occurrence data. *Ecology and Evolution*, *11*, 16177–16187.

Boyle, B., Hopkins, N., Lu, Z., Garay, J.A.R., Mozzherin, D., Rees, T., Matasci, N., Narro, M.L., Piel, W.H., Mckay, S.J., & others (2013) The taxonomic name resolution service: An online tool for automated standardization of plant names. *BMC Bioinformatics*, *14*, 16.

Chamberlain, S. (2016). scrubr: Clean biological occurrence records. R package version 0.1, **1**, 162.

Darrah, S. E., Bland, L. M., Bachman, S. P., Clubbe, C. P., & Trias-Blasi, A. (2017). Using coarse-scale species distribution data to predict extinction risk in plants. *Diversity and Distributions*, *23*, 435–447.

Dawson, T. P., Jackson, S. T., House, J. I., Prentice, I. C., & Mace, G. M. (2011). Beyond predictions: Biodiversity conservation in a changing climate. *Science*, *332*, 53–58.

de Jong, J., Gainer, M. & Russell, K. (2023) *Listviewer: "Htmlwidget" for interactive views of R lists*. https://CRAN.R-project.org/package=listviewer

Engemann, K., Enquist, B. J., Sandel, B., Boyle, B., Jørgensen, P. M., Morueta-Holme, N., Peet, R. K., Violle, C., & Svenning, J.-C. (2015). Limited sampling hampers "big data" estimation of species richness in a tropical biodiversity hotspot. *Ecology and Evolution*, *5*, 807–820.

Enquist, B. J., Condit, R., Peet, R. K., Schildhauer, M., & Thiers, B. M. (2016). *Cyberinfrastructure for an integrated botanical information network to investigate the ecological impacts of global climate change on plant biodiversity*. PeerJ Inc.

Enquist, B. J., Feng, X., Boyle, B., Maitner, B., Newman, E. A., Jørgensen, P. M., Roehrdanz, P. R., Thiers, B. M., Burger, J. R., Corlett, R. T., Couvreur, T. L. P., Dauby, G., Donoghue, J. C., Foden, W., Lovett, J. C., Marquet, P. A., Merow, C., Midgley, G., Morueta-Holme, N., … McGill, B. J. (2019). The commonness of rarity: Global and future distribution of rarity across land plants. *Science Advances*, *5*, eaaz0414.

Feng, X., Enquist, B. J., Park, D. S., Boyle, B., Breshears, D. D., Gallagher, R. V., Lien, A., Newman, E. A., Burger, J. R., Maitner, B. S., & Merow, C. (2022). A review of the heterogeneous landscape of biodiversity databases: Opportunities and challenges for a synthesized biodiversity knowledge base. *Global Ecology and Biogeography*, *31*, 1242–1260.

Feng, X., Merow, C., Liu, Z., Park, D. S., Roehrdanz, P. R., Maitner, B., Newman, E. A., Boyle, B. L., Lien, A., Burger, J. R., & Pires, M. M. (2021). How deregulation, drought and increasing fire impact Amazonian biodiversity. *Nature*, *1–6*, 516–521.

Fick, S. E., & Hijmans, R. J. (2017). WorldClim 2: New 1-km spatial resolution climate surfaces for global land areas. *International Journal of Climatology*, *37*, 4302–4315.

Franklin, J. (2010). *Mapping species distributions: Spatial inference and prediction*. Cambridge University Press.

Franklin, J., Serra-Diaz, J. M., Syphard, A. D., & Regan, H. M. (2017). Big data for forecasting the impacts of global change on plant communities: Big data for forecasting vegetation dynamics. *Global Ecology and Biogeography*, *26*, 6–17.

Gaston, K. J. (2003). *The structure and dynamics of geographic ranges*. Oxford University Press.

GBIF.Org User (2023) *Occurrence download*. https://doi.org/10.15468/dl.h3j2wj

Gotelli, N. J., Booher, D. B., Urban, M. C., Ulrich, W., Suarez, A. V., Skelly, D. K., Russell, D. J., Rowe, R. J., Rothendler, M., Rios, N., & Rehan, S. M. (2023). Estimating species relative abundances from museum records. *Methods in Ecology and Evolution*, *14*, 431–443.

Graham, C. H., Ferrier, S., Huettman, F., Moritz, C., & Peterson, A. T. (2004). New developments in museum-based informatics and applications in biodiversity analysis. *Trends in Ecology & Evolution*, *19*, 497–503.

Grenié, M., Berti, E., Carvajal-Quintero, J., Dädlow, G. M. L., Sagouis, A., & Winter, M. (2023). Harmonizing taxon names in biodiversity data: A review of tools, databases and best practices. *Methods in Ecology and Evolution*, *14*, 12–25.

Gueta, T., Barve, V., Nagarajah, T., Agrawal, A., & Carmel, Y. (2018). Introducing bdclean: A user friendly biodiversity data cleaning pipeline. *Biodiversity Information Science and Standards*, *2*, e25564. https://doi.org/10.3897/biss.2.25564

Guo, W.-Y., Serra-Diaz, J.M., Schrodt, F., Eiserhardt, W.L., Maitner, B.S., Merow, C., Violle, C., Anand, M., Belluau, M., Bruun, H.H.,, Byun, C., Catford, J. A., Cerabolini, B. E. L., Chacón-Madrigal, E., Ciccarelli, D., Cornelissen, J. H. C., Dang-Le, A. T., de Frutos, A., Dias, A. S., … Svenning, J.-C. (2022) High exposure of global tree diversity to human pressure. *Proceedings of the National Academy of Sciences*, *119*, e2026733119.

Guralnick, R. P., Hill, A. W., & Lane, M. (2007). Towards a collaborative, global infrastructure for biodiversity assessment. *Ecology Letters*, *10*, 663–672.

Jin, J., & Yang, J. (2020). BDcleaner: A workflow for cleaning taxonomic and geographic errors in occurrence data archived in biodiversity databases. *Global Ecology and Conservation*, *21*, e00852.

Kennedy, C. M., Oakleaf, J. R., Theobald, D. M., Baruch-Mordo, S., & Kiesecker, J. (2019). Managing the middle: A shift in conservation priorities based on the global human modification gradient. *Global Change Biology*, *25*, 811–826.

Lavoie, C. (2013). Biological collections in an ever changing world: Herbaria as tools for biogeographical and environmental studies. *Perspectives in Plant Ecology, Evolution and Systematics*, *15*, 68–76.

Lobo, J. M., Hortal, J., Yela, J. L., Millán, A., Sánchez-Fernández, D., García-Roselló, E., González-Dacosta, J., Heine, J., González-Vilas, L., & Guisande, C. (2018). KnowBR: An application to map the geographical variation of survey effort and identify well-surveyed areas from biodiversity databases. *Ecological Indicators*, *91*, 241–248.

Maitner, B., Gallagher, R., Svenning, J.-C., Tietje, M., Wenk, E. H., & Eiserhardt, W. L. (2023). A global assessment of the Raunkiæran shortfall in plants: Geographic biases in our knowledge of plant traits. *The New Phytologist*, *240*, 1345–1354.

Maitner, B. S., Boyle, B., Casler, N., Condit, R., Donoghue, J., Durán, S. M., Guaderrama, D., Hinchliff, C. E., Jørgensen, P. M., Kraft, N. J. B., McGill, B., Merow, C., Morueta-Holme, N., Peet, R. K., Sandel, B., Schildhauer, M., Smith, S. A., Svenning, J.-C., Thiers, B., … Enquist, B. J. (2018). The BIEN R package: A tool to access the Botanical Information and Ecology Network (BIEN) database. *Methods in Ecology and Evolution*, *9*, 373–379.

Maldonado, C., Molina, C. I., Zizka, A., Persson, C., Taylor, C. M., Albán, J., Chilquillo, E., Rønsted, N., & Antonelli, A. (2015). Estimating species diversity and distribution in the era of Big Data: To what extent can we trust public databases? *Global Ecology and Biogeography*, *24*, 973–984.

Merow, C. (2023) *occOutliers: Identify outlying species occurrence records*. R package version 0.1.0.

Meyer, C., Weigelt, P., & Kreft, H. (2016). Multidimensional biases, gaps and uncertainties in global plant occurrence information. *Ecology Letters*, *19*, 992–1006.

Panter, C. T., Clegg, R. L., Moat, J., Bachman, S. P., Klitgård, B. B., & White, R. L. (2020). To clean or not to clean: Cleaning open-source data improves extinction risk assessments for threatened plant species. *Conservation Science and Practice*, *2*, e311.

Pelletier, T. A., Carstens, B. C., Tank, D. C., Sullivan, J., & Espíndola, A. (2018). Predicting plant conservation priorities on a global scale. *Proceedings of the National Academy of Sciences*, *115*, 13027–13032.

Peterson, A. T., Soberón, J., Pearson, R. G., Anderson, R. P., Martínez-Meyer, E., Nakamura, M., & Araújo, M. B. (2011). *Ecological niches and geographic distributions (MPB-49)*. Princeton University Press.

Ribeiro, B. R., Guidoni-Martins, K., Tessarolo, G., Velazco, S. J. E., Jardim, L., Bachman, S. P., & Loyola, R. (2022). Issues with species occurrence data and their impact on extinction risk assessments. *Biological Conservation*, *273*, 109674.

Robertson, M. P., Visser, V., & Hui, C. (2016). Biogeo: An R package for assessing and improving data quality of occurrence record datasets. *Ecography*, *39*, 394–401.

Seebens, H., Clarke, D.A., Groom, Q., Wilson, J.R., García-Berthou, E., Kühn, I., Roigé, M., Pagad, S., Essl, F., Vicente, J., & others (2020) A workflow for standardising and integrating alien species distribution data. *NeoBiota*, *59*, 39–59.

Serra-Diaz, J. M., Enquist, B. J., Maitner, B., Merow, C., & Svenning, J.-C. (2017). Big data of tree species distributions: How big and how good? *Forest Ecosystems*, *4*, 30.

Serra-Diaz, J. M., Franklin, J., Ninyerola, M., Davis, F. W., Syphard, A. D., Regan, H. M., & Ikegami, M. (2014). Bioclimatic velocity: The pace of species exposure to climate change. *Diversity and Distributions*, *20*, 169–180.

Shaffer, H. B., Fisher, R. N., & Davidson, C. (1998). The role of natural history collections in documenting species declines. *Trends in Ecology & Evolution*, *13*, 27–30.

Sillero, N., Campos, J. C., Arenas-Castro, S., & Barbosa, A. M. (2023). A curated list of R packages for ecological niche modelling. *Ecological Modelling*, *476*, 110242.

Veiga, A. K., Saraiva, A. M., Chapman, A. D., Morris, P. J., Gendreau, C., Schigel, D., & Robertson, T. J. (2017). A conceptual framework for quality assessment and management of biodiversity data. *PLoS ONE*, *12*, e0178731.

Velazco, S. J. E., Rose, M. B., de Andrade, A. F. A., Minoli, I., & Franklin, J. (2022). Flexsdm: An r package for supporting a comprehensive and flexible species distribution modelling workflow. *Methods in Ecology and Evolution*, *13*, 1661–1669.

Zizka, A., Silvestro, D., Andermann, T., Azevedo, J., Duarte Ritter, C., Edler, D., Farooq, H., Herdean, A., Ariza, M., Scharn, R., Svantesson, S., Wengström, N., Zizka, V., & Antonelli, A. (2019). CoordinateCleaner: Standardized cleaning of occurrence records from biological collection databases. *Methods in Ecology and Evolution*, *10*, 744–751.

Zizka, A., Silvestro, D., Vitt, P., & Knight, T. M. (2021). Automated conservation assessment of the orchid family with deep learning. *Conservation Biology*, *35*, 897–908.

Zurell, D., Franklin, J., König, C., Bouchet, P. J., Dormann, C. F., Elith, J., Fandos, G., Feng, X., Guillera-Arroita, G., Guisan, A., Lahoz-Monfort, J. J., Leitão, P. J., Park, D. S., Peterson, A. T., Rapacciuolo, G., Schmatz, D. R., Schröder, B., Serra-Diaz, J. M., Thuiller, W., … Merow, C. (2020). A standard protocol for reporting species distribution models. *Ecography*, *43*, 1261–1277.

BIOSKETCH

Josep M. Serra-Diaz works on predicting plant distributions and vegetation dynamics under climate change. The author list team focuses on big data analysis for understanding biodiversity patterns and change.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Serra-Diaz, J. M., Borderieux, J., Maitner, B., Boonman, C. C. F., Park, D., Guo, W.-Y., Callebaut, A., Enquist, B. J., Svenning, J.-C., & Merow, C. (2024). occTest: An integrated approach for quality control of species occurrence data. *Global Ecology and Biogeography*, *00*, e13847. https://doi.org/10.1111/geb.13847