Raritas and RaritasVox: Programs for counting high diversity categorical data with highly unequal abundances

David Lazarus Corresp., [1] , **Johan Renaudie** [1] , **Dorina Lenz** [2] , **Patrick Diver** [3] , **Jens Klump** [4]

[1] Museum für Naturkunde, Berlin, Germany
[2] Leibniz-Institut für Zoo- und Wildtierforschung, Berlin, Germany
[3] Divdat Consulting, Wesley, Arkansas, United States
[4] CSIRO, Mineral Resources, Kensington, Australia

Corresponding Author: David Lazarus
Email address: david.lazarus@mfn.berlin

Acquiring data on the occurrences of many types of difficult to identify objects are often still made by human observation, e.g. in biodiversity and paleontologic research. Existing computer counting programs used to record such data have various limitations, including inflexibility and cost. We describe a pair of new open-source programs for this purpose - Raritas and RaritasVox, which share a similar graphical user interface for mouse based counting, and file output format. Raritas is written in Python and can be run as a standalone app for recent versions of either MacOS or Windows, or from the command line as easily customized source code. RaritasVox in addition supports voice based counting but is written in Java and is more complex to install or modify. Both programs explicitly support a rare category count mode which makes it easier to collect quantitative data on rare categories, e.g. rare species which are important in biodiversity surveys. Lastly, as to our knowledge no standards exist yet, we describe a new stratigraphic occurrence data (SOD) unitary file format which combines extensive metadata and a flexible structure for recording occurrence data of species or other categories in a series of samples.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018

Raritas and RaritasVox: programs for counting high diversity categorical data with highly unequal abundances

David B. Lazarus[1], Johan Renaudie[1], Dorina Lenz[2], Patrick Diver[3] and Jens Klump[4]

1 - Museum für Naturkunde - Leibniz-Institut für Evolutions- und Biodiversitätsforschung, Berlin, Germany
2 - Leibniz-Institut für Zoo- und Wildtierforschung, Berlin, Germany
3 - Divdat Consulting, Wesley, Arkansas, USA
4 - CSIRO, Mineral Resources, Kensington, Australia

Corresponding author - David Lazarus, david.lazarus@mfn.berlin

Author contributions

DBL created the main program specifications, designed the GUI and wrote the paper. JR wrote Raritas, DLenz and JK designed the voice functions and wrote RaritasVox. DBL and PD created the SOD format.

Abstract

Acquiring data on the occurrences of many types of difficult to identify objects are often still made by human observation, e.g. in biodiversity and paleontologic research. Existing computer counting programs used to record such data have various limitations, including inflexibility and cost. We describe a pair of new open-source programs for this purpose - Raritas and RaritasVox, which share a similar graphical user interface for mouse based counting, and file output format. Raritas is written in Python and can be run as a standalone app for recent versions of either MacOS or Windows, or from the command line as easily customized source code. RaritasVox in addition supports voice based counting but is written in Java and is more complex to install or modify. Both programs explicitly support a rare category count mode which makes it easier to collect quantitative data on rare categories, e.g. rare species which are important in biodiversity surveys. Lastly, as to our knowledge no standards exist yet, we describe a new stratigraphic occurrence data (SOD) unitary file format which combines extensive metadata and a flexible structure for recording occurrence data of species or other categories in a series of samples.

Introduction

Human observations as a source of scientific data

Quantitative data about many aspects of the natural world are collected in modern science with the use of instruments, but a substantial amount of observational data is still collected by human observation. This is particularly common in ecology, organismal biology and behavioral sciences, where the numeric data on the frequencies of occurrences of biologic phenomena are desired, but the objects/phenomena to be counted are too complex to identify by instruments or fully computerized image analysis systems. Up until the spread of desktop computers, such counts were done mostly either with the aid of mechanical counter buttons (including arrays of several buttons, to allow counting of multiple categories) or tallied by hand on printed list forms. Both methods are slow and require re-entering the count values into a computer afterwards before analysis, adding additional time and possibilities for error. Computer 'point-counting' programs can in principle replace these methods and at the same time provide additional functions that mechanical methods cannot, such as continuous statistical summaries of the data as it is being collected, which provides useful feedback to the observer on how complete or accurate the dataset being collected is.

Despite these obvious advantages counting programs have yet to fully replace manual methods. There are many reasons for this including cost, inflexibility, compatibility and inadequate ease of use. Numerous inexpensive or free simple tally counter programs are available that can replace mechanical counter buttons (e.g. dozens of simple smartphone/tablet apps, or more sophisticated desktop apps e.g. Versacount: (Kim & DeRisi, 2010). None of these however are well suited to counting larger numbers of categories, which is common in ecology, and in related fields such as paleontology. The need to count many objects in many categories is particularly acute in biodiversity related disciplines, e. g. field surveys of species diversity; species counts of fossil assemblages in micropaleontology. In such studies the diversity of objects and total numbers of objects available for study are both very high. Several programs have been developed to assist in biodiversity assessments (e.g. 'OrgaCount': www.aquaecology.de; 'Beecam': www.avansee.com). As many micropaleontologists work in commercial (oil industry) settings, there are also several sophisticated counting programs available (many as commercial products) for counting large numbers of microfossils: ; Polpal (Nalepka & Walanus, 2003); Foramsampler (Mcgann et al., 2006); Counter (Zippi, 2007); Stratabug (Stratadata, 2014); Bugwin (Bugware, 2016). These programs, whether for biologists or industrial micropaleontologists, however frequently are limited in one or more ways. Many are embedded in larger, more specialized packages with features for a single discipline, e.g. stratified ecologic sampling, biostratigraphic range charting, petrologic thin section analyses. Programs are often complex to install, or are lacking in flexibility, adaptability and/or ease of use. Many are also closed-source, expensive, and are dependent on the commercial provider to maintain. There is thus a need for a program that is relatively simple, free, open-source, less specialized and thus adaptable to counting a variety of different types of objects, and that works with different operating systems. Most importantly, it must be as easy to use as mechanical methods, since a program that is significantly slower will, based on our experience, normally be rejected by users. Users often need to count thousands of objects (see 'Rarity' below), and an even marginally slower data entry method will create an unacceptable cumulative loss of the user's time. This is particularly true in counting objects such as microfossils, or in field biodiversity surveys, where vast numbers of specimens are available and can be quickly identified by the user, making data entry the time-limiting factor in data

collection.

Rarity

In addition to the general need for flexible, efficient counting programs, there is also a specific need to count objects which have very different relative abundances. Many classes of objects in the observable world show a characteristic pattern of unequal relative abundances that can be approximated by power laws, including incomes, internet traffic, plankton sizes, and the sizes of interstellar mineral grains (Mathis et al., 1977, Reed & Hughes, 2002, Buonassissi & Dierssen, 2010). Biologic entities, in particular species abundances in ecology and paleontology also typically show such distributions, with a few species being relatively common, and the remainder uncommon or quite rare (Preston, 1948, Brown et al., 2002). Counting objects at random from such unevenly distributed populations results in many counts of the few common species, but very few counts of rarer species. For example, in both the complete dataset, and in individual samples, counts of fossil radiolarians in Neogene Southern Ocean sediments show a few very common species, and many rare species (Figs. 1, 2). Even with >700,000 individuals, a substantial fraction of the species are represented by 10 or fewer individuals. Thus, in order to encounter at least one individual of all rare species very large numbers of specimens need to be examined. For example, several thousand individuals needed to be examined in order to recover 95% of the estimated total species diversity (ca 200 species) in the single sample counted in Fig. 2 (Fig. 3).

Ecologists and paleontologists thus sometimes decide to base studies only on the small number of species that are relatively common and thus whose abundances are easy to quantify. Many applied micropaleontologic studies for example use the the environmental preferences of a relatively small number of common species to reconstruct past environmental conditions (Imbrie & Kipp, 1971, CLIMAP project members, 1976). Not all scientific questions can however be addressed by examination of only a small number of common species. Unlike, e.g. mineral grains, each biologic species is unique, with its own potential to contribute to ecosystem function and, over the longer term, to evolutionary change. Biodiversity research in particular is concerned about documenting total species richness and understanding threats to it, e.g. how current and past environmental change affects it. The findings of such research feed into important decisions on biodiversity conservation, land use and other global issues (i.e., the 'Rio' Convention on Biological Diversity: www.cbd.int). Reasonably accurate estimates of total diversity - crucial in biodiversity studies - can only be made when the majority of the diversity has been counted. Extrapolations from less complete data tend to have unacceptably high error values (Colwell et al., 2012). There is thus a major effort to understand the total species richness of modern and past biologic systems (Mora et al., 2011), and consequently, the need to collect quantitative data on many rare species (Roberts et al., 2016).

One approach to achieving this is based on the human ability to scan large populations to identify a subset of target individuals much more rapidly than the same person could fully identify and record the identity of each individual in the population. As a simple example, it is much faster to scan a large crowd of people to identify a single category of persons of interest ('tall men with beards'), than to identify each person in a crowd and record all of their names. Similarly, one can quickly skip individuals belonging to a specific category to target other individuals. Biologists and paleontologists collecting data on rare species make use of this ability by first counting all individuals encountered to identify common species, then, mentally blocking out the common species, continuing to count only species that are not in the 'common' group. In this 'rare category'

mode individuals of common species can be scanned over much more rapidly, and their counts for the total area viewed estimated afterwards based on their abundances in 'all species' mode. Larger total numbers of individuals are thereby examined, and a better estimate of total species richness can be obtained (Gannon, 1971, Hinds, 1999, Stevenson et al., 2010). A good counting program for such work should offer options that support this style of efficient counting of only rare taxa. This ability is however, to our knowledge, normally not offered in currently available counting programs, which are mostly designed to support counts of smaller numbers of species and individuals in support of applied (paleo)environmental research.

Materials and Methods

Raritas and RaritasVox are two new programs for counting (tallying) multiple categories of objects which meet these criteria. Both offer a flexible mouse-driven interface for counting highly diverse lists of taxa, including both buttons for more common taxa, and hierarchical menus to select rare taxa. An additional feature of the programs is the definition of a new file format for storing such count data that uniquely combines the data and detailed metadata in a user-friendly spreadsheet style layout. Compiled apps, source code, user guides, sample configuration and output files are all publicly available at https://github.com/plannapus/Raritas.

The programs provide explicit support of dual-mode (all vs rare only) counting, and indeed this feature is the basis for the program names. In standard mode, all individuals seen are counted. In 'rare only' mode, commonly occurring objects are no longer counted: only rare objects are. Not having to pause to enter a count for the most frequently seen object types makes counting rare object categories much faster. However, in order to be able to combine counts for common and rare types together, it is also necessary to know the magnitude of observational effort made in each counting mode, as the total frequencies of common objects are estimated for the 'rare objects only' interval based on their frequency in 'all object' counting, and the observational effort spent in 'rare' mode. A computer program that supports rare-only counting must therefore be able to monitor observational effort in parallel to recording individual object counts. This is provided for by a separate counter for observational effort, a 'track' counter which the user updates periodically while counting.

The main program Raritas, is written in Python (van Rossum et al., 2010). The second - RaritasVox - is written in Java, and was in fact the initial test development version. This older version provides most, though not all of the features of the main Python version in mouse-based counting. In addition it provides a unique option to register counts directly from voice input by the user, who simply speaks the category names. Regardless of method or program variant, the same type of output, setup and configuration files are used.

These programs' ease of use involve both ease of configuration as well as ease of use during primary operation. Raritas and RaritasVox are configured almost entirely from the contents of a simple tabular type file which can be created easily by users using a spreadsheet program. The file contains list of which objects (e.g. species) are to be counted, how these are to be presented to the user (button labels and other details). This also simplifies the program as there is no need to write code for configuration, other than reading the configuration file.

Detailed metadata is captured for each dataset and saved with the data in the output files. This

often a weakness in other (e.g. commercial) programs where relatively little information is captured. Reliance on program-external metadata capture such as embedding all metadata in filenames is obviously limited in extent, not well structured and in our experience has not been very reliable, particularly when metadata needs to be understandable over the long-term (i.e. by other than the file creators).

Raritas been programmed in Python because it is a popular, well supported, and relatively easy to learn multi-paradigm scripting computer language. It is more likely to be understandable to workers in fields such as taxonomy/systematics than the more complex, object-oriented compiled language Java. RaritasVox was programmed in Java in order to make use of specialized libraries for voice recognition: the Sphinx open-source speech recognition engine (Walker et al., 2004) (http://www.speech.cs.cmu.edu/sphinx/doc/Sphinx.html), and to insure speed, which is needed for the complex task of voice recognition - Java code executes much faster than Python code. Both programs run quickly on all hardware tested (desktop and laptop computers with Intel 'i' series processors, running Windows 7-10; OS X 10.9-12). Raritas consists of ca 650 lines of Python code; RaritasVox of nearly 4,000 lines of Java. The use of Python, plus the much smaller size of the code, makes customization of the Raritas's features possible by technically savvy users, without the need to employ a professional programmer. Python also provides excellent packages for some functions such as plotting data that allow the program to produce better outputs for the user without having to write additional code (e.g., matplotlib). Python is not without problems - installing the various software modules (packages), including packages used by other packages (dependencies) that an application needs can be very difficult for a non-specialist, depending in part on the local python environment used. Raritas is therefore offered both as a fully bundled program (double-clickable) with all needed packages included for Mac OS X 10.11+ as well as for Windows 7 and 10; and also as source code: the former providing ease-of-use for non specialists; the latter customizability. RaritasVox is also available either as a bundled app (a .jar file) or as source code. The bundled versions are each ca 100 Mb in size.

Installation

No special installation procedure is needed for the Raritas program when used as the bundled app. Using the source code version of Raritas (python) requires installing only two python packages (and their dependencies): matplotlib and wxPython (Hunter, 2007, Dunn, 2014). These must be installed using the appropriate python or OS package manager for the user's python system, which will automatically install any dependencies. Some python distributions already include both packages as part of their standard installation, thus requiring no special installations by the user. RaritasVox requires a Java environment (available for free download, often installed previously in many systems) in addition to the app itself. Installing the source code version of RaritasVox is considerably more complicated: details are given in Appendix 1.

Configuration file and starting the program

Both programs read a single configuration file on starting - by default, the one previously used, or a new one chosen by the user. The file (Fig. 4; Appendix 2) is in tab-text format and is just a list of taxa names and how each should be presented to the user in the GUI interface. All names are available by drop-down list by default. Names can also be shown as buttons (with abbreviations to insure the button label fits). If a second set of names of higher level categories are provided for the primary names, the name list is parsed into multiple list with multiple drop-down menus,

thus providing structure to longer name lists and more rapid access to taxa names.

Bundled versions of either program are started by the usual double-click of the app icon or other standard GUI methods. The source code version of Raritas is started by a standard 'python raritas.py' statement (optionally including a path name, if appropriate) at the command line. Once the program starts all interaction takes place via the GUI interface that then appears. RaritasVox cannot be run directly from the source code as Java is a compiled language - any customized version of the RaritasVox Java code must first be compiled and linked either via the command line or a programming tool such as an IDE.

GUI interface for manual counting

The main elements of the GUI interface for either version, once started, are: the metadata window, the counting window, the rare count configuration window and the collector curve window.

Metadata window (Fig. 5). When the program is first started a window appears which provides a pop-up list of primary counting style options (file types), based on the SOD file specification (described below). The next window collects the metadata appropriate for the file type, e.g. field names that are used in the rest of the program for the material to be counted. At the moment the program supports two types of primary data, both for microfossil occurrences: assemblages of microfossils from deep-sea sediments obtained by the international deep-sea drilling programs, or fossils from samples obtained from geologic sections on land, but other types can be defined. The metadata window also provides a few run-time options for configuring the interface and behavior during counting. Importantly, the user chooses which taxa name list configuration file they want to use via a normal file open dialog at this time. When ready the 'start counting' button is clicked and the counting window appears.

Counting window (Fig. 6). This is the main window that is used for most interaction with the program. The upper part of the window is populated with the buttons for counting common species, with labels as defined in the configuration file. Less common taxa are shown in the form of popup lists, organized into higher level categories, again as defined in the configuration file. Putting less common taxa into lists and common taxa on buttons allows most counts to be done quickly with a button, while the comparatively slow process of selecting from a list is reduced to a minimum. Lists are needed however as they can be of arbitrary length, while the number of buttons is limited by screen size. Counting is active whenever the window is present. Clicking on a button or selecting a taxa from the lists adds the species to the count data structures. A list of recently counted objects is given in the sub-window (lower middle of main window). A button is provided on the right to count observational effort ('Track', for number of 'tracks' scanned on a microscope slide') and a counter shows the total tracks counted.

Clicking on 'Rare Count Mode' brings up a dialog (Fig. 7), where the counted objects are listed in order of descending abundance, and the user can choose which to exclude from further counting. When the dialog is dismissed counting resumes, with, for those taxa to be excluded, the taxa buttons greyed out and pop-up list items inactivated.

Determining which species to exclude in rare count mode is not trivial. As this is a key feature of Raritas we include the following suggestions, which are based on our experience of counting ca

700,000 total specimens (several thousand specimens per sample in over 100 samples) for the study published in (Renaudie & Lazarus, 2013). The tally to use to trigger the switch to rare-only counting, and the percentage threshold for species to be ignored during 'rare' count mode should, as a rule of thumb, maximize the number of specimens to ignore while minimizing the error on the abundant species percentages. In (Renaudie & Lazarus, 2013), we chose to stop the full count mode when ca. 2,000 specimens were already counted and to ignore in 'rare' count mode species with a percentage higher than ~5% of the community. Doing so allowed us to keep the error to ca. 10% of the investigated value. In other words, for a species that was present at 5% abundance in full count mode, the theoretical standard error is slightly below 10% of this 5% value, i. e. a theoretical percentage for the species between ca. 4.5 and ca. 5.5%; (Drooger, in (Zachariasse et al., 1978) (Fig. 8a). These cut-off values eliminated 59.7% of the specimens during rare-only mode (median of all samples counted, but varying from one sample to the other, black line on Fig. 8b for median, dark grey area for interquartile range and light grey are for total range). An additional, important criterion that was taken into consideration is that all samples encountered had at least one species above the 'ignore in rare-only mode' percent threshold. Using an higher threshold than 5% would have meant that some samples would have had to be counted entirely in full count mode, as no species would have been abundant enough to exclude. In our study, there were on average ca three (mean = 2.9) percent of the species above the cut-off threshold per sample (blue and red lines of Fig. 7b).

The 'Show Collector's Curve' menu item (Raritas, or button, RaritasVox) brings up the fourth main GUI element - a diversity accumulation plot (Fig. 9) showing the relationship to total number of object types seen (species) vs total number of objects counted (specimens). For typical biologic data these curves show a roughly logarithmic in shape - at first rising rapidly, then, as increasingly species already seen previously are re-encountered, flattening out. The curve's slope will eventually become zero when all object types in the sample have been detected (compare to Fig. 2). The user can decide when the curve has become close enough to this state for his/her purposes, and thus stop counting only when the data completeness quality is adequate. If a series of samples are counted to the point where they have the same apparent slope at the end of this dynamically generated diversity accumulation curve, they will share the property of being 'fairly' sampled, and relative differences in diversity will be shown without bias (Alroy, 2010, Colwell et al., 2012). This type of feedback is important to insuring good quality observations and is something that cannot be provided by simple mechanical count systems. It is however rarely implemented in programs known to us.

Voice interface

RaritasVox has a similar GUI to Raritas, with only fairly minor differences in the layout of elements or functional behavior (e.g., RaritasVox allows colors to be assigned to taxa names as an aid to accurate name selection in the interface), and thus is not described separately here - details are given in Appendix 1. The main difference in functionality is the ability to use a voice driven counting mode, selected via a control button from the main counting window. The motivation was the observation that, for some users, the constant change of focus between microscope and counting program (or paper sheet) while counting microfossils under a microscope places a strain on the user's vision. Some researchers affected by this problem had developed a voice-based counting procedure: calling out species identifications and recoding the counts as audio recordings, then later playing them back and transferring the species counts into their counting sheets. RaritasVox was conceived as a way, by using speech recognition, to make this process

more efficient and ergonomic.

Since 2009 when RaritasVox was developed and today speech recognition has made tremendous advances and and has become a commonplace functionality in many everyday applications, e.g. Apple's "Siri". Speech recognition systems can be classified into two categories. "Speaker dependent" systems use "training" (also called "enrollment") where an individual speaker reads text or isolated vocabulary into the system. The system analyzes the person's specific voice and uses it to fine-tune the recognition of that person's speech, resulting in increased accuracy. Systems that do not use training, including RaritasVox, are called "speaker independent" systems. RaritasVox however makes use of the fact that the counting process uses an independent vocabulary that is defined in a configuration file (Fig. 10; Appendix 2). The user may not only use his or her own short terms for species rather than the full taxonomic name, e.g. "pachyleft" instead of "*Globigerina pachyderma* sinistral", they can modify the configuration file so that the program can better recognize an individual's normal pronunciation style. This is for example useful for users with different native languages, as vowels in particular are often pronounced differently, even for latin taxa names. For example "*Prunopyle*" is pronounced proo-no-peil by English speakers, and proo-no-peel-ae by Germans.

At the time RaritasVox was first being planned (2009) only a few cross-platform packages were available. The speech recognition software Sphinx and Java were chosen as the best combination for an open-source, cross platform speech recognition package and language environment for our purposes. For Sphinx the elemental components of speech sounds are interchangeably referred to as "phones" or "phonemes" (see http://www.speech.cs.cmu.edu/sphinx/doc/Sphinx.html and http://www.speech.cs.cmu.edu/cgi-bin/cmudict). Only phonemes listed in the phoneme set of the CMU Pronouncing Dictionary (around 40) can be used and it expects that the language used is English. Only words consisting of one or more phonemes that are present in the customized dictionary file (Fig. 10) can be recognized as "correct". The software will search for words consisting of phonemes present in the dictionary which match best to the speech input. In RaritasVox the spoken word is recognized, confirmation is shown on screen, and a count command for that item is generated (Fig. 11).

RaritasVox was not used to collect research data and was only briefly tested for accuracy (Table 1).

Using a list of 18 words and 108 voice entries, four words were incorrectly identified (<4%), resulting in 8 incorrect counts (7.5%). This is similar to accuracy in much more sophisticated, general voice recognition systems [27], which is possible as RaritasVox uses a very limited vocabulary. The count error rate may be too large for data collection where rare occurrences are important (e.g. biostratigraphy) but adequate for others such as gross assemblage composition, particularly when combined with statistical data reduction procedures such as factor analysis that are insensitive to small amounts of random data scatter [13]. The accuracy is in any event choosable by the user as they can, by monitoring the computer screen, correct errors before they are counted using the spoken 'Remove' command to delete the last (incorrect) identified word.

Output files

SOD File Format

In addition to the diversity accumulation plots, which can be saved as graphics as often as desired (the matplotlib library used in Raritas supports various file formats, e.g. png, pdf, jpg, tif), the program saves the primary count data. This necessitates choosing, or creating a format for the data files, as there is no universal community database which would allow a direct upload solution. Despite a great deal of biostratigraphic or other data of the form of species by samples/observations having been generated globally for many decades, no generally accepted or even widely known file format exists for such data. Other fields have developed community data formats for such data matrices, e.g. the BIOM format for biological observation matrices (McDonald et al., 2012), as well as standard protocols to exchange information directly between computer systems e.g. Darwin Core (Wieczorek et al., 2012). These formats are however of limited use for paleontologic fossil occurrence matrices since they lack any way to store metadata, general or individual sample, that is related to geologic age (sample position in section, formation name, etc), and the metadata in general is optimized for biologic, not paleontologic observations. One of the major biologic exchange protocols (ABCD: (Berendsohn, 2007), http://wiki.tdwg.org/ABCD/) does have, via the EFG extension (http://www.geocase.eu/efg) the ability to transmit both biologic and geologic data, but is a communication protocol, not storage format, and the xml definition is not readable by normal users.

Within the field of paleontology, data on occurrences, outside of micropaleontology, are dominated by simple taxa lists for a single locality (one sample). This is exemplified by the main data input formats the most widely used paleontology community database PBDB (Alroy et al., 2001), where data is entered, taxon by taxon, for one sample at a time. Within micropaleontology taxa-by-sample data matrices are common (often referred to as 'range charts') but data is usually given in the format of individual publications, without metadata in the files, in numerous variations of a simple taxa-by-sample table. This is also the file format used by the deep-sea drilling programs (DSDP, ODP, IODP), which have not generally captured micropaleontology data except in a very limited form on-ship, using database entry forms, or simply archived data copied from publications, with only minimal metadata stored separately from the data files. Lastly there are several more comprehensive data file formats that are associated with commercial micropaleontology, i.e. the oil industry. These formats include metadata, details of stratigraphy etc, but are not compatible with each other and are mostly meant for internal use in proprietary commercial programs, not for open file exchange. Most also tend to be quite user unfriendly, giving sample and taxa names in separate definition blocks from the actual occurrence data, and use a long, non-tabular, list type structure that makes comprehension difficult. There is thus a need for a public (non-proprietary) file format that combines metadata and the taxa-by-occurrences data in a single file, provides for geologic age or section information and which is easy for scientists to read and use.

We have therefore adopted a new 'open file format': Stratigraphic Occurrence Data format, which we abbreviate here simply as SOD format. This format originally was developed in response to the need to merge metadata and occurrence data in user typed files, in order to manage a large number of fossil occurrence matrix files that were being digitized from the literature for upload into a database that provides a micropaleontologic equivalent to the PBDB: NSB (Lazarus, 1994, Spencer-Cervato, 1999). This database reports occurrences of microfossils in deep-sea sediment sections, and the data is mostly derived from studies that report the occurrences in the form of simple samples by species tables, one table per section, per higher fossil group. The file format itself is deliberately meant to be visually similar to the source publication data tables, being essentially an enhanced version of the publication's tabular data matrix. This makes the file easily read by users, and equally makes the transcription (keying-in) of data from publications into the

format relatively simple - in some cases, where a publication file is available in digital form, simply by reformatting some of the fields, rather than re-entry of primary values. SOD format however is significantly different from an 'ordinary' user data table in that it is based on a formal, extendable definition of content. This definition adds more structure and detail for both taxa and sample names, and uses the otherwise empty 'corner' of the matrix at the intersection of the row and column labels to include, in a structured way, more general metadata about the occurrence data in the file.

The file is laid out in 4 graphical blocks: general metadata: upper left corner block; taxa metadata: left columns below metadata block; sample metadata: rows to right of corner metadata block; and the occurrence data itself in the remaining lower right block (Fig. 12). Flexibility is provided for in two ways. The individual fields in each block can be populated by different actual data types, depending on the overall record type as determined by the 'File Type' field. Currently there are only two defined file types, for deep-sea drilling data and more traditional land section data (O and L, respectively). These differ both in general metadata (Site location vs geographic name and geographic coordinates), and in the way in which sample names are structured: deep-sea drilling samples ('O' files) use a consistent Site-Hole-Core-Section-Interval format, while land sections are more variably defined, but usually include some combination of geologic formation, vertical position in section and sample name (usually unique to each study); with additional information often recorded on geologic age or biostratigraphic zone and lithology. SOD 'L' formatted files include all these fields. Within the broad constraints on total fields available, the number of file types using this layout is open to indefinite expansion. The SOD layout itself is also extensible, as the version is written in the first metadata field in each file. The field definitions and thus the data expected in each field are determined by these control fields, and different layouts can be defined, for example with additional rows for sample name fields. This flexibility however requires a separate source of information that defines, for the user and programmer, what the field contents must be for each 'File Type' or SOD version number. These definition requirements are the fundamental difference between regular data files as found in the literature, and the SOD format. The definitions are given in two ways (which also allows cross checking for data consistency). First, the tabular file definition requires full labeling - each cell, row or column that holds data has an adjacent cell with fixed text content defining the data cell(s) adjacent, so that the content resembles a simple key:value non-relational database structure. This means the files are largely self documenting, and provides sufficient explanatory information to users so that they can create new data files from a template file (containing labels but no data values). Second, programs that read SOD files are expected to have a definition table of some sort which gives the location and meaning of each cell for each file type and each SOD version. Currently this is implemented in a table in the NSB database and used by programs (both a python script and an R procedure at present) that read and upload SOD data into the NSB system. This definition list could also be included (e.g. as a second 'page' in a spreadsheet file) with the data files themselves. A full list of current SOD field definitions and additional details on the format are given in Appendix 3.

Over 500 files have been created in SOD format, both typed or edited by users as described above, or generated by the Raritas program during counting of microfossils. Raritas generates only data for one sample at a time, but otherwise the output is identical to that used for complete sample by taxa matrices in other SOD files. SOD formatted files are not intended to replace more complex, formally controlled, computer-to-computer data exchange formats, defined in xml or other systems. SOD is best viewed as complementary, providing a user accessible format that encourages the capture of the metadata needed to adequately document stratigraphic occurrence

data, which until now has often not been done. It should also be noted that the SOD format is much more flexible and can accommodate many more types of data than the current versions of Raritas programs themselves, which are 'hard wired' to work e.g. with Taxa and Sample Names. Future versions of these programs ideally should be modified to read the fields needed for the metadata window, and output data file formats, directly from a SOD definition file.

Diversity vs number of specimens

The program outputs, in addition to the main count data, the cumulative diversity vs number of counted objects history as a simple tab-text data file. This data can be useful for fitting rarefaction curves in subsequent data analyses.

Results

The degree to which biodiversity assessments can be improved using our software depends on a variety of factors - the distribution of taxon abundances (evenness) and absolute diversity of the target population(s) being counted; and the ability of the user to mentally mask out taxa and focus only on those not excluded. Most people can easily keep a 'skip' list of several taxa in mind when counting, but not a much larger list, e.g. a dozen or more taxa. Thus the improvement in counting with Raritas tends to be best when the abundances are significantly uneven and the total diversity is less than a few hundred categories. In the example shown in Figures 1 and 7 of this paper, from Antarctic Pleistocene radiolarian assemblages, by eliminating the 6 most common species (cumulative abundance of >74% of the specimens in the sample) nearly 3/4 of the specimens can be skipped, allowing an effective sampling of the rarer taxa that is 4X what would have been possible by counting all specimens. In practice we have found that we more typically increase our effective sample size by 2-3X by using rare count mode. These increased effective sample sizes significantly improve the accuracy of diversity estimates, although the precise amount will depend on total sample size, evenness and absolute diversity (Colwell et. al., 2012).

Discussion and Conclusions

The programs described here provide useful tools for counting populations with large numbers of categories and unequal abundances of individuals in categories. They are, as programmed, best suited to micropaleontology studies, but with only minor modification can be adapted to many other uses in biodiversity research and other fields. The SOD definition provides a flexible, internally documented yet easy to read file format for storing and exchanging occurrence data, either for individual populations or matrices with multiple sets of observations. The Raritas program described here has proved itself in actual use over several years in the junior author's research group in Berlin. As noted above, it has been used to count >700,000 specimens belonging to several hundred different species in >100 radiolarian microfossil assemblages, as part of a study of biodiversity change in the Southern Ocean over the last 20 my (Renaudie & Lazarus, 2013). It has been used by several individuals in other projects including students, on a variety of computers.

Acknowledgements

The authors wish to thank the numerous individuals for the open-source software tools used to create the Raritas programs.

References

Stevenson RJ, Pan Y, van Dam H. 2010. Assessing environmental conditions in rivers and streams with diatoms, p. 57–85. In Smol JP, Stoermer EF (ed), *The Diatoms: Applications for the Environmental and Earth Sciences*, Cambridge University Press, Cambridge.

Alroy J, Marshall CR, Bambach RK, Bezusko K, Foote M, Fürsich FT, Hansen TA, Holland SM, Ivany LC, Jablonski D, Jacobs DK, Jones DC, Kosnik MA, Lidgard S, Low S, Miller AI, Novack-Gottshall PM, Olszewski TD, Patzkowsky ME, Raup DM, Roy K, Sepkoski JJ, Jr., Sommers MG, Wagner PJ, Webber A. 2001. Effects of sampling standardization on estimates of Phanerozoic marine diversification. *Proceedings of the National Academy of Sciences (USA)* 98:6261–6266.

Alroy J. 2010. Fair sampling of taxonomic richness and unbiased estimation of origination and extinction rates, p. 55–80. In Alroy, J, Hunt G (ed), *Quantitative Methods in Paleobiology*, The Paleontological Society,

Berendsohn W. 2007. Access to Biological Collection Data. ABCD Schema 2.06 - ratified TDWG standard. TDWG Task Group on Access to Biological Collection Data, BGBM, Berlin http://www.bgbm.org/TDWG/CODATA/Schema/default.htm.

Brown JH, Gupta VK, Li BL, Milne BT, Restropo C, West GB. 2002. The fractal nature of nature: power laws, ecological complexity and biodiversity. *Phil Trans R Soc* 357:619–626.

Bugware. 2016. Bugwin. http://www.bugware.com

Buonassissi CJ, Dierssen HM. 2010. A regional comparison of particle size distributions and the power law approximation in oceanic and estuarine surface waters. *Journal of Geophysical Research* 115:C10028 (1–12).

CLIMAP members. 1976. The surface of the ice-age earth. Science 191:1131–1137.

Colwell RK, Chao A, Gotelli NJ, Lin S-Y, Mao CX, Chazdon RL, Longino JT. 2012. Models and estimators linking individual-based and sample-based rarefaction, extrapolation and comparison of assemblages. *J Plant Ecol* 5:3–21.

Dunn R. 2014. wxPython, version. 3.0. wxpython.org.

Gannon JE. 1971. Two counting cells for the enumeration of zooplankton micro-crustacea. *Trans Am Micros Soc* 90:486–490.

Hinds WC. 1999. *Aerosol Technology: Properties, Behavior, and Measurement of Airborne Particles, 2nd Edition*, Wiley, Hoboken, NJ.

Hunter JD. 2007. matplotlib: A 2D graphics environment. *Computing in Science and Engineering* 9:90–95.

Imbrie J, Kipp NG. 1971. A new micropaleontological method for quantitative paleoclimatology: application to a late Pleistocene Carribean core, p. 71–181. In Turekian KK (ed), *Late Cenozoic Glacial Ages*, Yale University Press, New Haven.

Kim CC, DeRisi JL. 2010. VersaCount: customizable manual tally software for cell counting. *Source Code Biol Med* 5:web.

Lazarus DB. 1994. The Neptune Project - a marine micropaleontology database. *Math Geol* 26:817–832.

Mathis JS, Rumpl W, Nordsieck KH. 1977. The size distribution of interstellar grains. *The Astrophysical Journal* 217:425–433.

McDonald D, Clemente JC, Kuczynski J, Rideout JR, Stombaugh J, Wendel D, Wilke A, Huse S, Hufnagle J, Meyer F, Knight R, Caporaso JG. 2012. The Biological Observation Matrix (BIOM) format or: how I learned to stop worrying and love the ome-ome. *GigaScience* 1:1–6.

Mcgann M, Mcgann LB, Bonomassa O, Devries P, Luther J, Malmberg S, Nelson G, Pratt SIII.

2006. Foramsampler v. 3.0 - microfossil sample data management software. *Anuário do Instituto de Geociências* 29:278–279.

Mora C, Tittensor DP, Adl SM, Simpson AGB, Worm B. 2011. How many species are there on earth and in the ocean? *PLoS Biology* 9:1–8 (web).

Nalepka D, Walanus A. 2003. Data processing in pollen analysis. Acta Paleobot 43:125–134.

Preston FW. 1948. The commonness, and rarity, of species. *Ecology* 29:254–283.

Reed WJ, Hughes BD. 2002. From gene familes and genera to incomes and internet file sizes: why power-laws are so common in nature. *Physical Review* E 66:67103–67106.

Renaudie J, Lazarus D. 2013. On the accuracy of paleodiversity reconstructions: a case study in Antarctic Neogene radiolarians. *Paleobiology* 39:491–509.

Roberts TE, Bridge TC, Caley MJ, Baird AH. 2016. The Point Count Transect Method for Estimates of Biodiversity on Coral Reefs: Improving the Sampling of Rare Species. *PLoS One* 11:e0152335.

Spencer-Cervato C. 1999. The Cenozoic deep sea microfossil record: explorations of the DSDP/ODP sample set using the Neptune database. *Palaeontologica Electronica* 2:web.

Stratadata. 2014. Stratabugs biostratigraphic data management software. http://www.stratadata.co.uk

van Rossum G, Drake J. 2010. Python Language Reference, version 2.7.

Walker W, Lamere P, Kwok P, Raj B, Singh R, Gouvea E. 2004. Sphinx-4: a Flexible Open Source Framework for Speech Recognition, Sun Microsystems, Mountain View, CA.

Wieczorek J, Bloom D, Guralnick R, Blum S, Döring M, Giovanni R, Robertson T, Vieglais D. 2012. Darwin Core: An evolving community-developed biodiversity data standard. *PlosOne* 7:1–8.

Zachariasse WJ, Riedel WR, Sanfilippo A, Schmidt RR, Brolsma MJ, Schrader HJ, Gersonde R, Drooger MM, Broekman JA. 1978. Micropaleontological counting methods and techniques-an exercise on an eight metres section of the lower Pliocene of Capo Rossello, Sicily. *Utrecht Micropaleontological Bulletins* 17:79-176.

Zippi P. 2007. Counter 4.5. PAZ Software. http://www.pazsoftware.com.

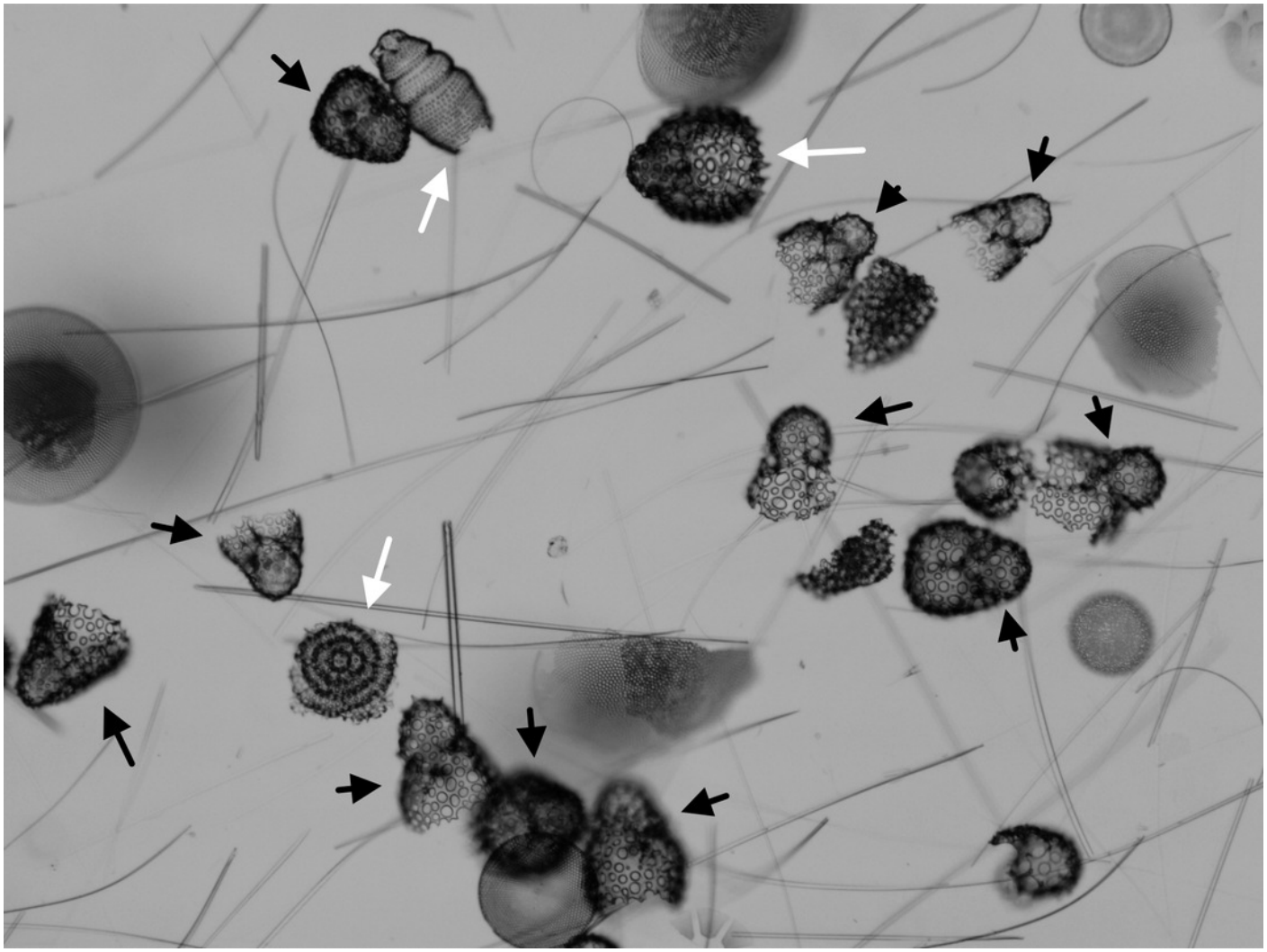
Supporting Information Appendices

S1 Appendix 1 - User Guides
S2 Appendix 2 - Sample Files
S3 Appendix 3 - SOD Definition

Figure 1

Assemblages with common and rare taxa

Microfossil assemblage as seen in the microscope (late Pleistocene, Southern Ocean, ODP Site 751). Specimens marked by black arrows all belong to *Antarctissa strelkovi* or *A. denticulata*. Other radiolarian species are marked by white arrows. Unmarked individuals are not targets for counting - broken radiolarians and diatom valves. Most individuals in this target assemblage belong to just a few species (particularly *A. strelkovi* and *A. denticulata*), making discovery of rarer taxa difficult.



PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018

Figure 2(on next page)

Ranked relative abundances of fossil radiolarian species in single samples and combined multisample datasests.

Counts of species, sorted by abundance, of Neogene Southern Ocean radiolarian assemblages, showing total dataset (several dozen samples) and a single sample (Deep-sea drilling sample ODP 751A-6H-6, 98-100 cm). Despite a total count of 7071 specimens within the single sample, the majority of the species are represented by 6 or fewer individuals. From data in (Renaudie & Lazarus, 2013) SOM.

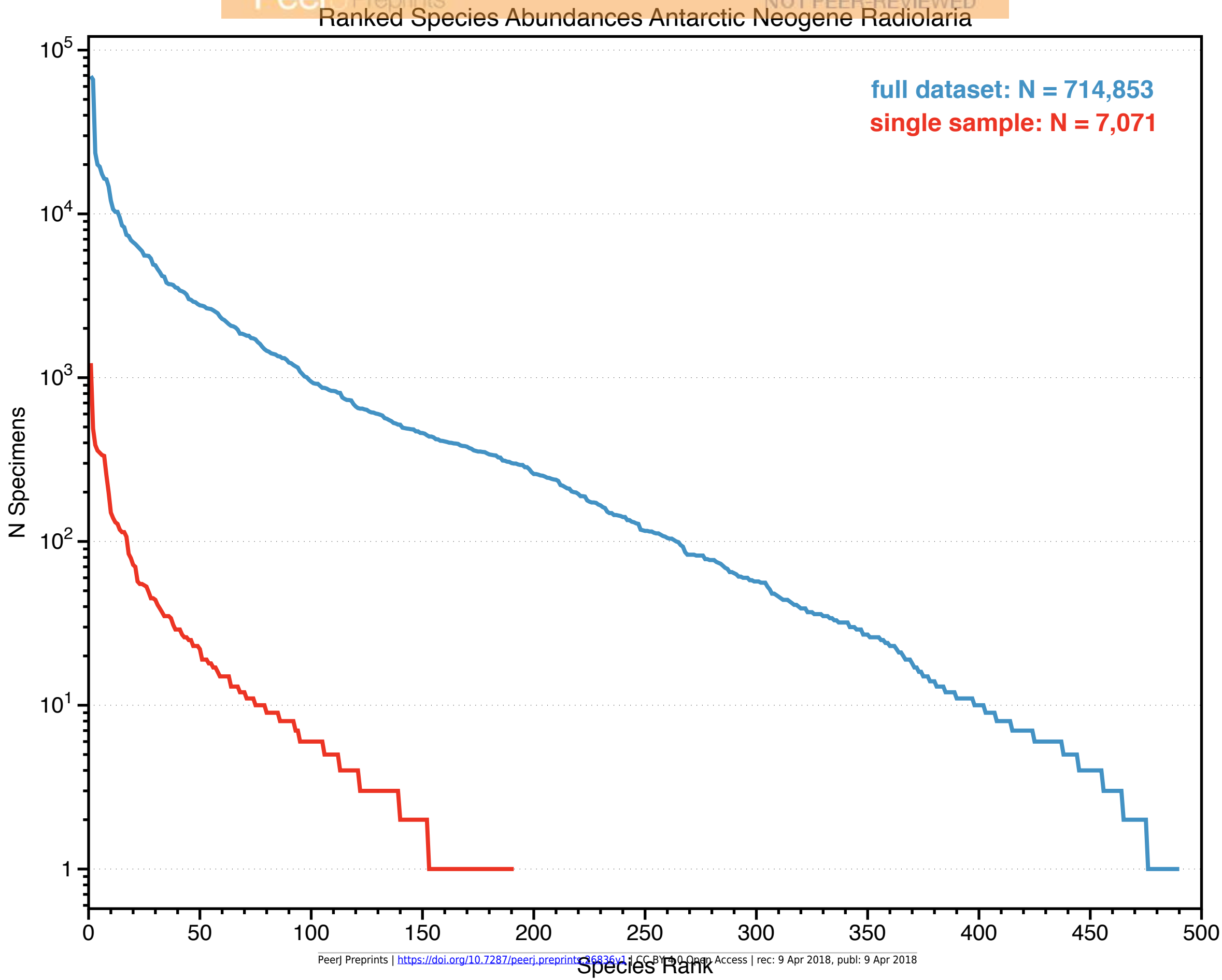


Figure 3(on next page)

Cumulative diversity vs sample size curve and estimated true diversity for a single sample.

Species-accumulation curve on a typical sample (sample ODP 751A-6H-6, 98-100 cm shown in Fig 1). Bold black curve is the species accumulation curve; light grey curve is a de Caprariis type curve-fit; dashed light grey line its asymptote (i.e. species diversity at infinite sample size). From (Renaudie & Lazarus, 2013).

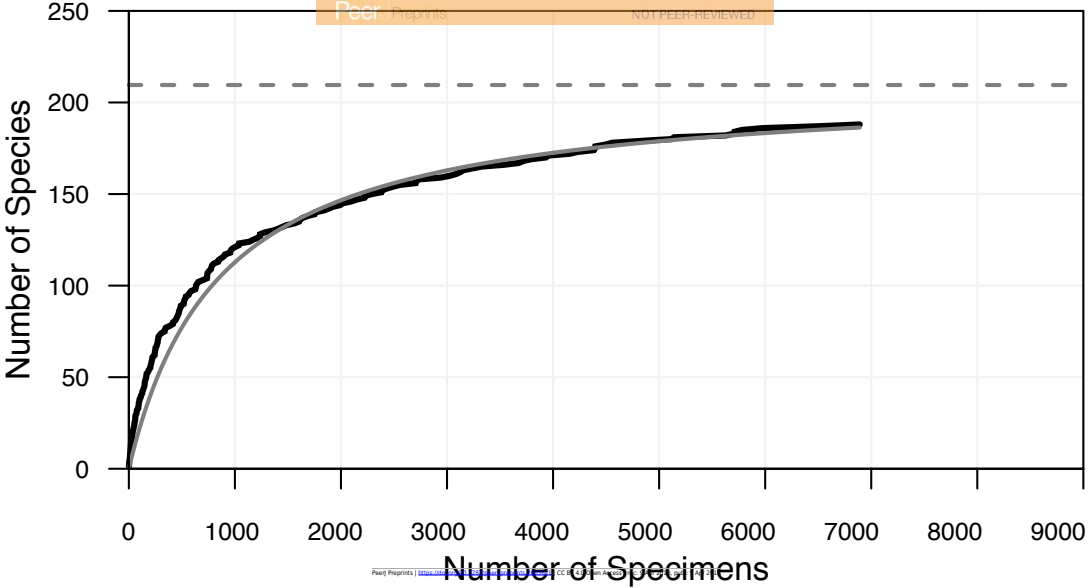


Figure 4

Configuration file to populate interface with category names.

Configuration file format (a plain text file, here formatted for easier reading). Only a few fields - 'Genus' and 'Species' components of a taxonomic name, button (yes/no) are mandatory. A couple fields, e.g. 'Recognition Name' are used only by RaritasVox.

Genus	GQ	Species	SQ	Sub-species	Author	HigherTaxon	Comment	Color	Button	abbreviation	Recognition Name	listNr.
Acanthodesmia		micropora				Nassellaria			n	Aca mic		2
Acanthosphaera		actinota				Spumellaria			n	Aca act		3
Acanthosphaera		insignis				Spumellaria			n	Aca ins		3
Acrosphaera		australis				Collo/Entact/Phaeo			y	australis		1
Acrosphaera		cyrtodon				Collo/Entact/Phaeo			n	Acr cyr		1
Acrosphaera		labrata				Collo/Entact/Phaeo			y	labrata		1
Acrosphaera		lappacea				Collo/Entact/Phaeo			n	Acr lap		1
Acrosphaera		mercurius				Collo/Entact/Phaeo			n	Acr mer		1
Acrosphaera		murrayana				Collo/Entact/Phaeo			y	murrayana		1
Acrosphaera		cuniculiauris				Collo/Entact/Phaeo			y	cuniculi		1
Acrosphaera		spinosa				Collo/Entact/Phaeo			n	Acr spi		1
Actinomma		arcadophorum				Spumellaria			n	Act arc		3
Actinomma		boreale				Spumellaria			n	Act bor		3
Actinomma		campilacantha				Spumellaria			n	Act cam		3
Actinomma		delicatulum				Spumellaria			y	Act del		3
Actinomma		golownini				Spumellaria			y	golownini		3
Actinomma		eldredgei				Spumellaria			n	Act spE		3
Actinomma		kerguelensis				Spumellaria			n	Act ker		3

Figure 5

Dialog to enter general sample metadata.

Metadata window used for Raritas. Information about the sample to be counted is entered here, including observer, date, class of objects being counted ('Fossil Group'), and sample identification information. RaritasVox has additional options (not shown), e.g. 'Save list of counted species with diversity' which, if checked, creates a second output file that gives the entire history of counting.

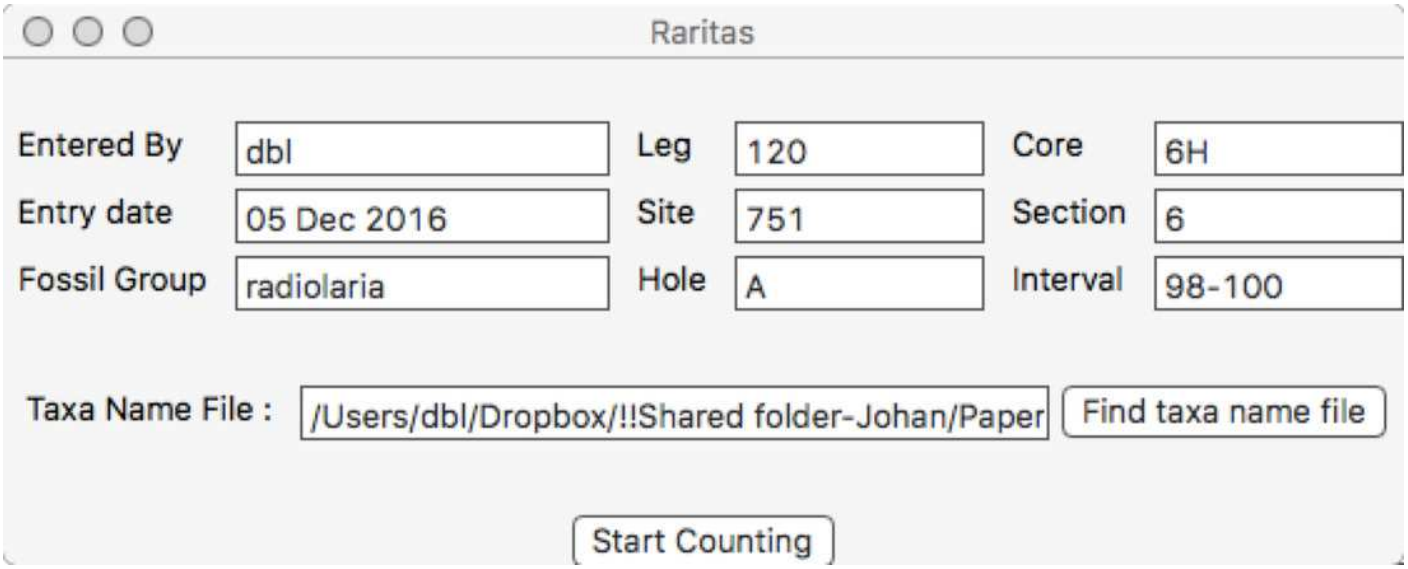


Figure 6

Main counting window with buttons, hierarchical category menus and count status information.

Main counting window. Objects to be counted are presented in two forms: an array of clickable buttons in the upper part of the window, and as a set of pop-up lists in the lower left and center part of the window. The number of lists and their contents is automatically built from the configuration file higher category labels for object entries. Button labels are also taken from this file on start-up. Other buttons or menu items control program behavior and call up other features e.g. voice recognition (RaritasVox only), show count plot, switch to Rare Count mode etc. A scrolling list of the most recently counted objects is shown in the lower middle. The 'Track' counter and clickable (large rectangular) button are on the lower right and are used to record observation effort in both regular and rare count modes. Note, in this image rare count mode has already been activated; thus some buttons are greyed out.

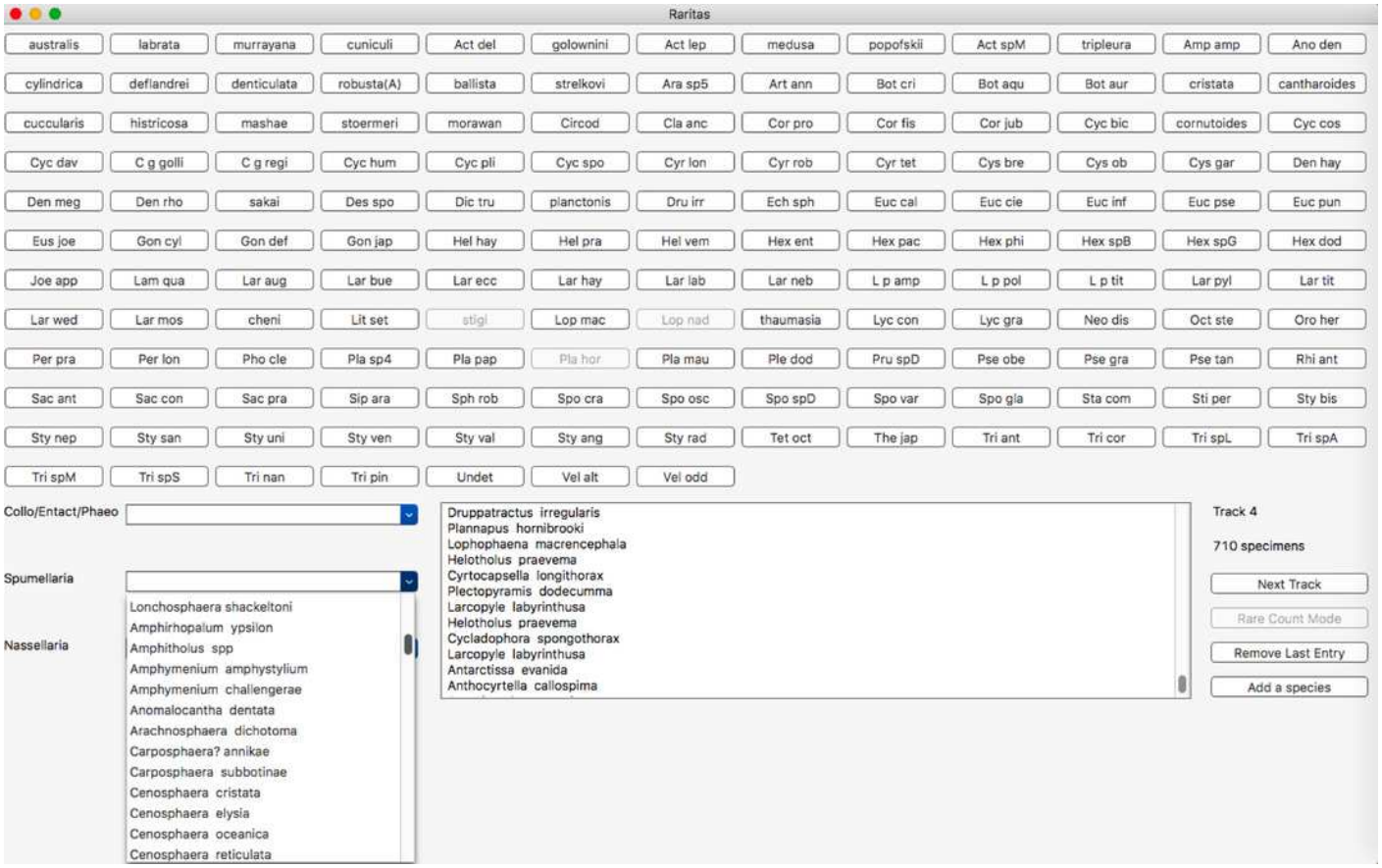
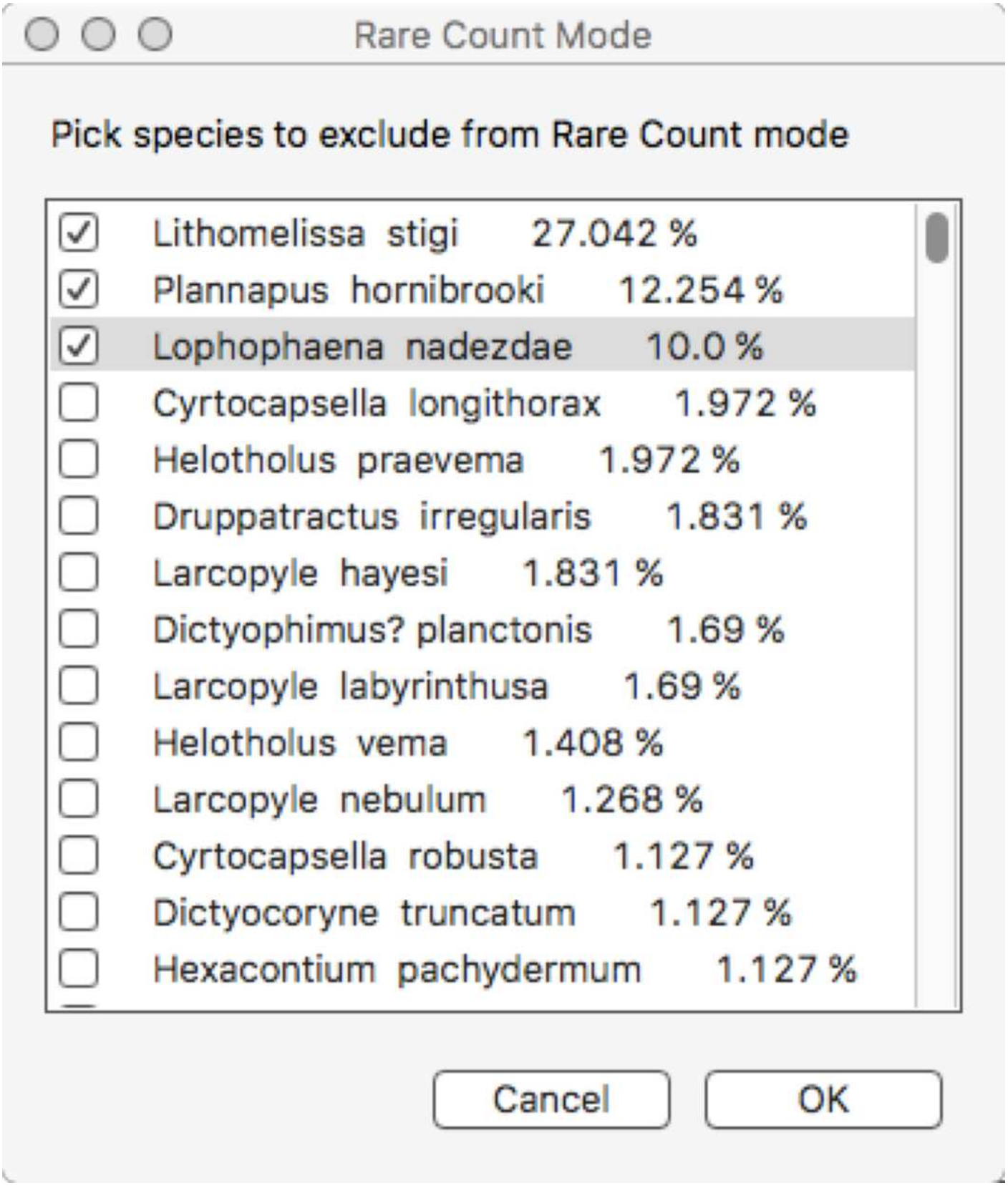


Figure 7

Dialog to configure rare count mode.

Configure rare count mode dialog. The object counts list, sorted by count frequencies, is presented and the user selects those objects (here, species names) that will in skipped and no longer counted in rare count mode.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018



PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018

Figure 8(on next page)

Relationships between sample size and uncertainty of abundance estimates in generalized and actual biodiversity data.

Panel A (left) - Epsilon (size of confidence interval, relative to the abundance value, for a given species relative abundance in a population) plotted on a p (percent) vs N (number of specimen) landscape. Rule of thumb used in [12] marked by dashed lines (Renaudie & Lazarus, 2013) highlighted. Panel B (right) - Shows, for data reported in (Renaudie & Lazarus, 2013), red line: the percent of samples that have at least one species with percent higher than p; blue line: the percent of species having a proportion higher than p in at least one sample, and black line with shading: the cumulative proportion of specimens of species with proportion higher than p (mean, inner-quartile range and total range over all 107 samples).

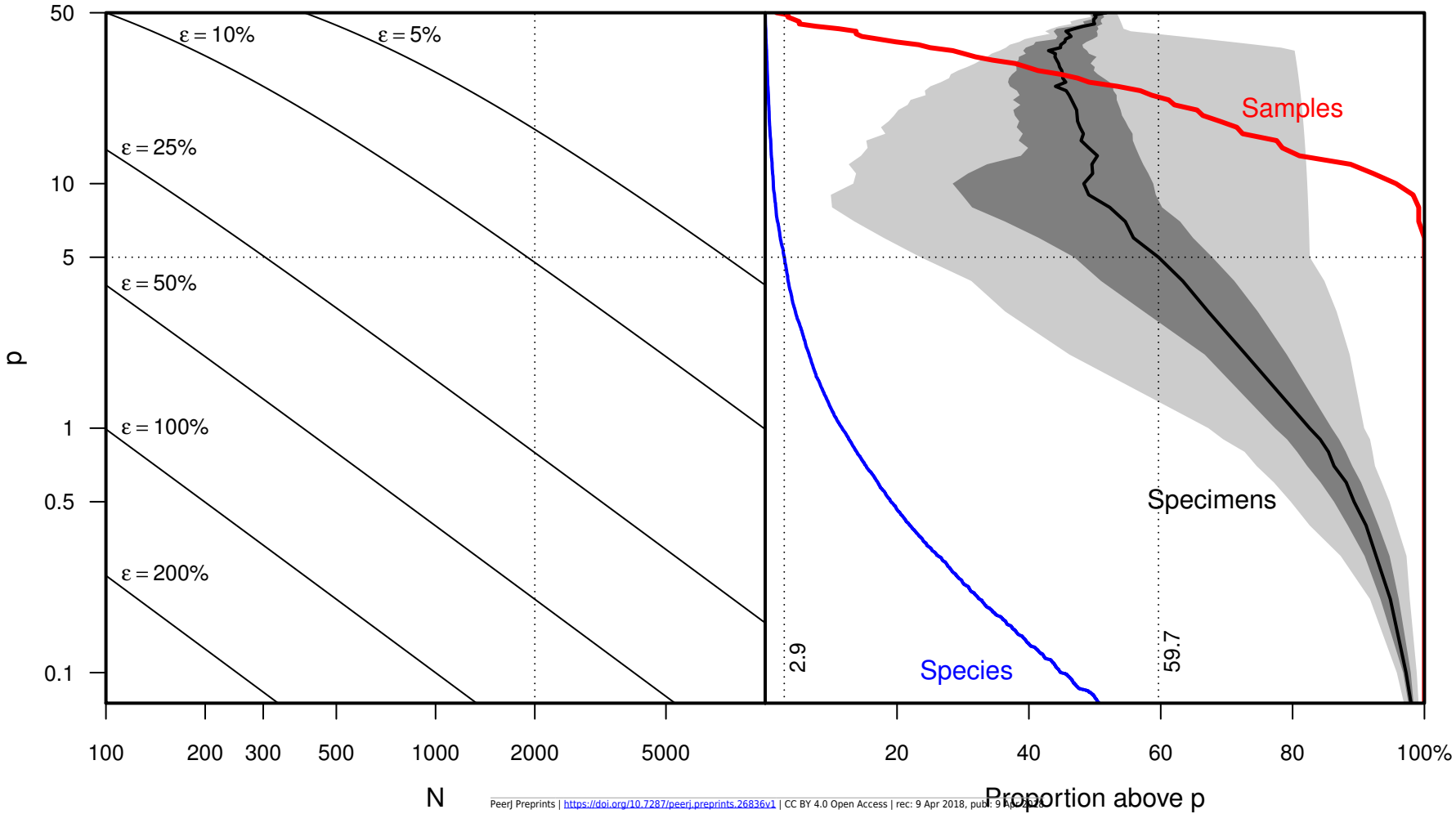
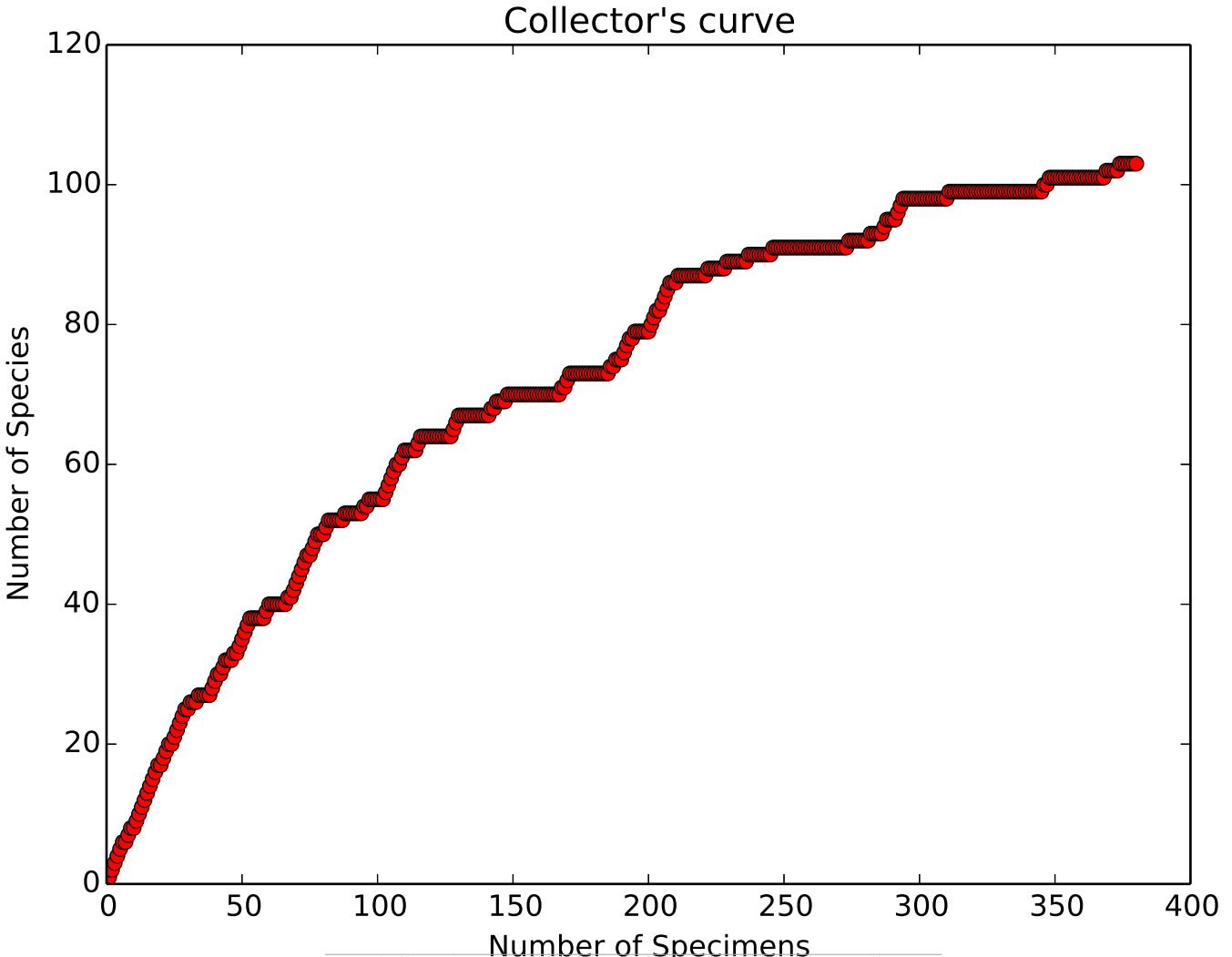


Figure 9(on next page)

Collecting curve, showing history of cumulative diversity vs sample size.

Count plot window, showing a simple graphic of how total diversity of objects ('species') is increasing with increased numbers of counted objects ('specimens'). The window appears whenever the user clicks the 'show count plot' button in the main counting window. This graphic is calculated and plotted anew with each invocation. The shape of the curve provides important feedback for the user, see text for details.



PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018

Figure 10(on next page)

RaritasVox defined vocabulary and pronunciation configuration file.

Configuration file for voice recognition using RaritasVox (extract only). Spoken words are on the left and the phoneme pronunciations on the right.

APROX	AH P R OW K S
STOCKI	S T OW K IY
AMPRADIOSA	AE M P R AE D IY OW S AH
ARTANNULATUS	AA R T AE N AH L EY T AH Z
AXIRREGULAR	AE K S IH R EH G Y AH L ER
BGRAN	B IY G R AE N
CRYPTBUSS	K R IH P T B AH S
GONDWANA	G AO N D W AE N AH
LOPHOHADRA	L OW F AH HH AE D R AH
MITA	M IY T AH
PODPAPILIS	P AA D P AE P IH L IH S
PSEUDODICT	S UW D OW D IH K T
ZYGO	S IY G ER
SPYRO	S P IY R ER
CORNUTELLA	K AO R N Y UH T EH L AH
CALOCYCLAS	K AE L OW S AY K L AH Z
BUNNYEARS	B AH N IY IH R Z
ZIGZAG	Z IH G Z AE G

Figure 11

Main counting window for RaritasVox

Screenshot of RaritasVox in voice-counting mode. A list of acceptable words is shown in the top window, the currently recognized word in large letters in the middle of the screen (to make it easy to see at a glance when e.g. working at a microscope), button controls below this and summary panes of count activity at the bottom.

Figure 12(on next page)

Example of SOD file format with data blocks framed.

Example of SOD file output (the main data output file produced by Raritas), with the 4 main areas (blocks) marked by bold lines. Metadata about the data file is stored in the upper left block, object labels and linked data such as author names, if known, are in the lower left block, sample information is in the upper right block, and the actual counting data in the lower right block. In output from the Raritas program only a single column of data is created but the SOD format definition permits the sample name and count values to repeat indefinitely (to the right of this figure). Note that only a few selected rows are shown here - the full file has ca 400 taxa names.

count for ms fig

SOD v.:	2.1	File Type:	O	Fossil Group:	radiolaria					
Source ID:		Source Name:		Source Citation:					Site:	751
Entered By:	dbl	Entry Date:	18-01-2017	Checked By:		Check Date:			Hole:	A
Leg Info:	120	Leg Qualifier:							Core:	12H
				File Creation Method:	Raritas				Section:	2
Occurrences Data Type:	C	Keys:							Interval top:	12-14
Comments:	1 tracks observed								Depth(mbsf):	
									Abundance:	A
									Preservation:	G
Genus:	GQ:	Species:	SQ:	Subspecies:	Author:	Taxon Code:	Higher Taxon:	Taxon Comments:		
Acrosphaera		murrayana					Collo/Entact/Phaeo			3
Acrosphaera		spinosa					Collo/Entact/Phaeo			1
Actinomma		golownini					Spumellaria			13
Anomalocantha		dentata					Spumellaria			1
Antarctissa		strelkovi					Nassellaria			18
Antarctissa		ballista					Nassellaria			8
Botryostrobus		auritus/australis					Nassellaria			1
Cycladophora		humerus					Nassellaria			8
Cycladophora		golli		regipileus			Nassellaria			2
Cycladophora		golli		golli			Nassellaria			1
Dendrospyris	?	sakaii					Nassellaria			3
Dendrospyris		rhodospyroides					Nassellaria			2
Dictyophimus	?	planctonis					Nassellaria			14
Druppatractus		irregularis					Spumellaria			8

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018

Table 1 (on next page)

Recognition accuracy in a simple test run of RaritasVox.

Accuracy of spoken entry using RaritasVox for a short list of species name abbreviations. Each name was spoken in random order 6 times. Note the independence of the spoken and data names e.g. zigzag for *L. robusta*. The spoken and formal names are linked in the Vox configuration file.

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26836v1 | CC BY 4.0 Open Access | rec: 9 Apr 2018, publ: 9 Apr 2018

Genus	GQ	Species	SQ	spoken name	VOX count	Errors
Amphicraspedum		prolixum	gr.	aprox	5	1
Amphipyndax		stocki		stocki	6	0
Amphisphaera		radiosa		ampradiosa	6	0
Artostrobus		annulatus		artannulatus	6	0
Axoprunum		irregularis		axirregular	5	1
Buryella		granulata		bgran	6	0
Calocyclas		spp.		calocyclas	7	1
Cornutella		sp.		cornutella	6	0
Cryptocarpium		bussonii	gr.	cryptbuss	8	2
Gondwanaria	?	sp.		gondwana	6	0
Lithomelissa		robusta		zigzag	6	0
Lophocyrtis		hadra		lophohadra	5	1
Acrosphaera		cuniculiauris		bunnyears	6	0
Mita	?	sp.		mita	6	0
Podocyrtis		papilis		podpapilis	5	1
Pseudodictyophimus		gracilipes		pseudodict	6	0
Spyrocyrtis		A	n.sp.	spyro	6	0
Zygocircus		buetschli		zygo	7	1
					108	**8**