ECOGRAPHY

Software note

canaper: Categorical analysis of neo- and paleo-endemism in R

Joel H. Nitta [1], Shawn W. Laffan [2], Brent D. Mishler [3] and Wataru Iwasaki [1,4,5]

[1]Department of Integrated Biosciences, Graduate School of Frontier Sciences, The University of Tokyo, Chiba, Japan
[2]Earth and Sustainability Science Research Centre, School of Biological, Earth and Environmental Sciences, The University of New South Wales, Sydney, NSW, Australia
[3]University and Jepson Herbaria, and Department of Integrative Biology, University of California Berkeley, Berkeley, CA, USA
[4]Department of Biological Sciences, Graduate School of Science, The University of Tokyo, Tokyo, Japan
[5]Atmosphere and Ocean Research Institute, The University of Tokyo, Chiba, Japan

Correspondence: Joel H. Nitta (joelnitta@gmail.com)

Biodiversity has typically been quantified using species richness, but this ignores evolutionary history. Due to the increasing availability of robust phylogenies, methods have been developed that incorporate phylogenetic relationships into quantification of biodiversity. CANAPE (categorical analysis of neo- and paleo-endemism) is one such method that can provide insight into the evolutionary processes generating biodiversity. The only currently available software implementing CANAPE is Biodiverse, which is written in Perl and can be used either through a graphical user interface (GUI) or user-developed scripts. However, many researchers, particularly in the fields of ecology and evolutionary biology, use the R programming language to conduct their analyses. Here, we present canaper, a new R package (www.r-project.org) that provides functions to conduct CANAPE in R. canaper implements methods for efficient computation, including parallelization and encoding of community data as sparse matrices. The interface is designed for maximum simplicity and reproducibility; CANAPE can be conducted with two functions, and parallel computing can be enabled with one line of code. Our case study shows that canaper produces equivalent results to Biodiverse and can complete computations on moderately sized datasets quickly (<10 min to reproduce a canonical study). canaper allows researchers to conduct all analyses from data import and cleaning through CANAPE within R, thereby obviating the need to manually import and export data and analysis results between programs. We anticipate canaper will become a part of the toolkit for analyzing biodiversity in R.

Keywords: biodiversity, CANAPE, canaper, endemism, phylogeny, R, spatial phylogenetics

www.ecography.org

Introduction

Quantifying biodiversity is a major goal of ecology. The simplest and most commonly used biodiversity metric is species richness (Diamond 1975). However, as all taxa are related to some degree by descent from a common ancestor, a thorough understanding of biodiversity is only possible by considering their evolutionary relationships. This became possible with the development of phylogenetic measures of biodiversity, such as phylogenetic diversity (PD; Faith 1992) and phylogenetic endemism (PE; Rosauer et al. 2009) (Box 1). Such analyses are becoming much more common due to the widespread availability of robust molecular phylogenies and large spatial datasets (Thornhill et al. 2017, Mishler et al. 2020, Nitta et al. 2022).

One recently developed extension of PE is categorical analysis of neo-and paleo-endemism (CANAPE; Mishler et al. 2014). CANAPE uses phylogenetic methods to give insight into evolutionary processes that produce centers of endemism (areas with many narrow-ranged taxa; or in the phylogenetic sense, concentrations of narrow-ranged branches of the phylogeny). In theory, centers of endemism may arise via multiple processes. For example, previously widespread lineages may undergo extinction in all but a portion of their range, leading to paleo-endemism. Alternately, recently diverged lineages may only occur in a small area and lead to neo-endemism. It is also possible that a given area is home to a high concentration of both paleo- and neo-endemic lineages (mixed endemism). CANAPE involves analyzing observed patterns of PE in comparison with a null model to first infer whether an area is significantly high in PE, and then to classify those areas into centers of paleo-endemism, neo-endemism, or mixed endemism. CANAPE is widely used (>50 publications in Google Scholar query for papers that cite Mishler et al. (2014) and mention 'CANAPE'), and is a central component of the field of spatial phylogenetics.

Despite the popularity of CANAPE, it has so far only been implemented in one software package, Biodiverse (Laffan et al. 2010). Biodiverse is written in Perl and comprises an analytical engine and a graphical user interface (GUI). While Biodiverse is convenient for non-coders because of its GUI, many ecologists and evolutionary biologists use R (https://www.r-project.org) for their analyses (Lai et al. 2019). Until now, an R user who wanted to conduct CANAPE analysis as part of a broader R workflow needed to first clean raw data, export it to Biodiverse, conduct PD and PE analyses in Biodiverse, then import the results back into R for further analysis and visualization. Also, prior to ver. 4, Biodiverse calculated the metrics needed to categorize endemism type but did not actually perform this categorization. Furthermore, the Biodiverse GUI alone does not currently support parallel processing, which is important for large datasets. Parallel processing and automation of Biodiverse analyses is only currently possible using Perl scripts. A set of R scripts is available to call Biodiverse Perl scripts from R (https://github.com/NunzioKnerr/biodiverse_pipeline), but it is not an R package and does not conduct CANAPE within R.

Here, we present a new R package that implements CANAPE completely in R: canaper. We strove to make canaper simple to use and efficient. Parallel computing can be enabled with a single line of code. canaper has passed code review meeting the 'silver' standards for statistical software at rOpenSci (Boettiger et al. 2015), and is verified against a large number of unit tests (>99% coverage). All results are reproducible by setting the random seed generator in R, in both sequential and parallel computing modes.

Box 1: Glossary of terms.

1) Community: A co-occurring set of species.
2) Grid-cell: One of a set of spatial units each with equal area that together cover a region of interest; each grid-cell comprises a community. Often used synonymously with 'site'.
3) Community matrix: A 2-dimensional dataset with communties (grid-cells) on one axis and species on the other. canaper expects communities as rows and species as columns.
4) Phylogenetic diversity (PD): The total branch length (including the root) connecting the species occurring in a given area (community or grid-cell), measured on an overall tree that includes all the species in the study (Faith 1992).
5) Phylogenetic endemism (PE): Range weighted PD where the weight for each branch is the fraction of its overall range represented in the sample area (typically a grid-cell) (Rosauer et al. 2009).
6) Relative phylogenetic diversity (RPD): The ratio of PD measured on the original tree vs. PD measured on a comparison tree with all branch lengths transformed to equal length (Mishler et al. 2014).
7) Relative phylogenetic endemism (RPE): The ratio of PE measured on the original tree vs. PE measured on a comparison tree with all branch lengths transformed to equal length (Mishler et al. 2014).
8) Neoendemic: An area with a high concentration of short, range-restricted branches; may be due to processes such as recent speciation (radiation).
9) Paleoendemic: An area with a high concentration of long, range-restricted branches; may be due to processes such as extinction or colonization by distantly relatives from outside the study area.
10) Mixed endemic: An area with a mixture of short and long, range-restricted branches; may be due to multiple processes.
11) Super endemic: An area with mixed endemism that is statistically highly significant.

Installation

The stable version of `canaper` can be installed from CRAN, and the latest development version from r-universe (https://r-universe.dev/).

```
# Stable version
install.packages("canaper")
# Development version
install.packages("canaper", repos =
"https://ropensci.r-universe.dev")
```

Input data format

Community data

Community data is provided as a data frame or matrix, with species as columns and communities (also referred to as 'sites' or 'grid-cells') as rows. Metrics of phylogenetic diversity including CANAPE can be measured at any taxonomic level, but for simplicity we use 'species' here. In this case, the data must include both row names and column names. Community data may also be input as a tibble, in which case site names must be indicated in a dedicated column (default column name `"site"`), rather than row names since tibbles lack row names. Community data may be either presence-absence data (0 or 1 s) or abundance data (integers $>=0$). However, calculations of PD and PE for CANAPE use only presence-absence information (i.e. no abundance weighting is used), so identical results will be obtained whether the input data is abundance or abundance that has been converted to presence-absence. Community data is typically loaded using `read.csv()`, `readr::read_csv()`, or other functions that can import rectangular data. The `points2comm()` function of the `phyloregion` package may be used to convert raw occurrence data (e.g. latitude and longitude of species occurrences) to a community matrix (Daru et al. 2020).

Phylogeny

The `ape` R package (https://CRAN.R-project.org/package=ape) is used to handle phylogenies, which are stored as lists of the class `"phylo"`. Phylogenies are required to have branch lengths $>=0$, but are not required to be fully bifurcating. Phylograms, chronograms, or cladograms may be used. Phylogenies can be loaded with the `ape::read.tree()` function.

Analysis workflow

The entire CANAPE workflow can be run with two functions, `cpr_rand_test()` and `cpr_classify_endem()`. However, internally this entails several steps that the user should be aware of as follows (Fig. 1).

Calculate observed values

First, the input phylogeny is scaled so the sum of all branch lengths is 1 and observed phylogenetic diversity (`pd_obs`) and phylogenetic endemism (`pe_obs`) are calculated (Box 1). Next, an alternative phylogeny (also referred to as the 'comparison tree') is constructed that has non-zero branch lengths set to a constant value (`canaper` arbitrarily uses 1), then rescaled so the sum of all branch lengths is 1. PD and PE are then calculated on the alternative phylogeny (`pd_alt`, `pe_alt`). Relative PD and PE, the ratio of `pd_obs` to `pd_alt` and `pe_obs` to `pe_alt` respectively, are then calculated (RPD, RPE; Box 1). However, the statistical
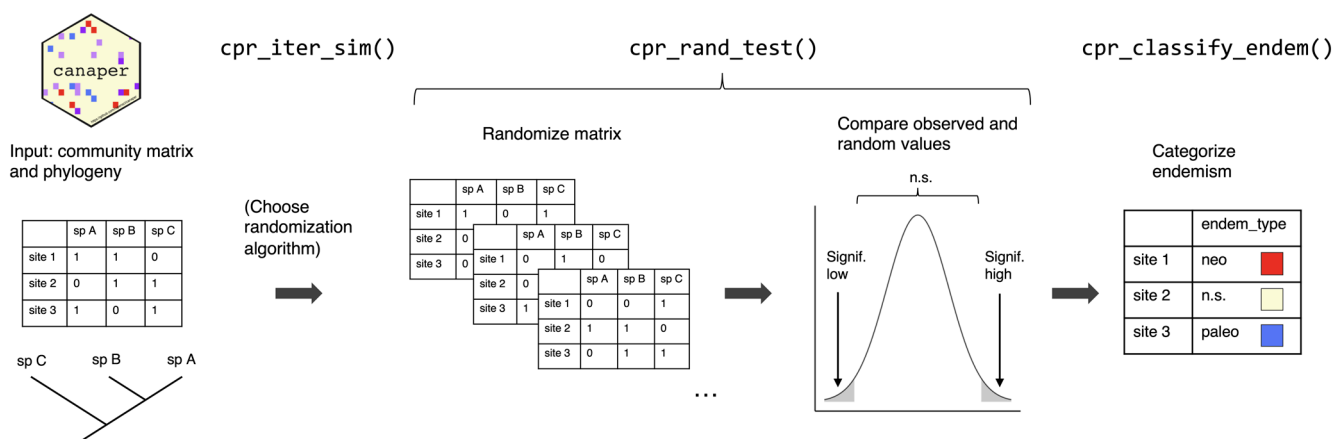

Figure 1. `canaper` workflow and functions to conduct categorical analysis of neo- and paleo-endemism (CANAPE). Input data are a community matrix and phylogeny. First, the user selects a randomization algorithm and its settings, with optional assistance from the `cpr_iter_sim()` function. Next, a set of random communities are generated, various metrics (e.g. phylogenetic endemism, relative phylogenetic endemism) are calculated for the input community and each random community, and the rank (p-value) of each observed metric is compared to those of the random communities by the `cpr_rand_test()` function. Finally, endemism types are categorized on the basis of the ranked metrics with the `cpr_classify_endem()` function. For details on metrics and classification, see 'Analysis workflow'. For details on functions, see 'Major functions'.

significance of any of these metrics cannot be determined from observed values alone.

Generate random communities

PD (and by extension, PE) will increase with taxon richness, since adding a new taxon samples more branches of the tree. To determine the statistical significance of the calculated metrics, the observed value is compared to a distribution of values obtained from a set of random communities. The random communities are generated by a randomization algorithm that shuffles the original data.

Since the randomization algorithm influences the range of reference (expected) values, the choice of randomization algorithm is likely to have a large effect on the results. As there is no single 'correct' algorithm, we have opted to provide the user with a wide range of options by supporting randomization algorithms included in the `vegan` package (https://cran.r-project.org/package=vegan). `vegan` includes > 30 randomization algorithms, but not all are appropriate for CANAPE. Recommended algorithms include `swap` (Gotelli and Entsminger 2003) and `curveball` (Strona et al. 2014). These algorithms preserve the number of sites occupied by each species and the richness of each site, and both produce results comparable to one commonly used randomization algorithm in Biodiverse called `rand_structured` (see 'Example: Australian *Acacia*'). `curveball` has been shown to be more computationally efficient than `swap`, but we are unaware of any studies comparing the effects on PD or CANAPE between the two.

We have also provided a method for users to provide a custom, user-defined randomization algorithm using the `vegan` framework. This may be appropriate if, for example, the community matrix includes a very wide area and it is desired to restrict randomizations to subsets of the area.

Calculate summary statistics

Once a randomization algorithm has been selected, random communities are generated for a number of replicates set by the user, and a set of summary statistics are computed (Supporting information). Summary statistics include the mean and standard deviation of PD, RPD, PE and RPE of the random communities and comparisons of observed values to the random communities including standard effect size and rank, which is used to calculate p-values.

Categorize endemism

The final step in CANAPE is to categorize endemism as described in Mishler et al. (2014). Briefly, this is done by comparing significance values of summary statistics calculated in the previous step (Supporting information). To be considered significantly endemic, a given grid-cell must first have significantly high `pe_obs` or `pe_alt` or both (one-tailed test). If this is true, the grid-cell is classified into one of three non-overlapping categories: if the grid-cell has significantly high

or low RPE (two-tailed test), it is considered to be a center of paleo-endemism or neo-endemism, respectively; if RPE is not significant (but `pe_obs`, `pe_alt`, or both are), it is considered a center of mixed endemism. Centers of mixed endemism can be further divided based on p-value; if `pe_obs` and `pe_alt` and both significant at the $\alpha = 0.01$ level, the grid-cell may be considered a center of super-endemism (but not all studies make this distinction).

Major functions

`cpr_rand_comm()`

The `cpr_rand_comm()` function generates a single random community. The first argument, `comm`, is a community data frame (or matrix). The second, `null_model`, is the name of one of the predefined randomization algorithms to use. The remainder of the arguments are particular to specific types of randomization algorithm. `cpr_rand_comm()` is typically not called by the user directly, but is provided to help users select randomization algorithms and settings.

One feature to be aware of is that randomization algorithms in `vegan` are classified as either binary or quantitative. Binary algorithms are designed for binary (i.e. presence-absence) data, and quantitative algorithms are designed for quantitative (i.e. abundance) data. Either type of algorithm will accept either type of data, but binary algorithms will convert abundance data to binary and return a binary matrix (data frame).

As the calculations of PD and PE in `canaper` do not take into account abundance (i.e. no abundance weighting is used), identical results will be obtained by either using abundance data or converting abundance data to binary before analysis. In this sense, the binary randomization algorithms are appropriate for CANAPE.

The following code illustrates use of `cpr_rand_comm()` with a set of example data that comes with `canaper`, the test data from Phylocom (Webb et al. 2008).

```
library(canaper)

# Set a seed for reproducibility
set.seed(12345)

# The original data includes abundance
data(phylocom)
phylocom$comm
     sp1 sp10 sp11 sp12 sp13 sp14 sp15
sp17 sp18 sp19 sp2 sp20 sp21 sp22
clump1 1 0 0 0 0 0 0 0 0 0 1 0 0 0
clump2a 1 2 2 2 0 0 0 0 0 0 1 0 0 0
clump2b 1 0 0 0 0 0 2 2 2 2 1 2 0 0
clump4 1 1 0 0 0 0 0 2 2 0 1 0 0 0
even 1 0 0 0 1 0 0 1 0 0 0 0 1 0
random 0 0 0 1 0 4 2 3 0 0 1 0 0 1
```

```
        sp24 sp25 sp26 sp29 sp3 sp4 sp5
sp6 sp7 sp8 sp9
clump1 0 0 0 0 1 1 1 1 1 1 0
clump2a 0 0 0 0 1 1 0 0 0 0 2
clump2b 0 0 0 0 1 1 0 0 0 0 0
clump4 0 2 2 0 0 0 0 0 0 0 1
even 0 1 0 1 0 0 1 0 0 0 1
random 2 0 0 0 0 0 2 0 0 0 0
# Binary null model produces binary
output
cpr_rand_comm(phylocom$comm, "swap", n_
iterations = 100)
       sp1 sp10 sp11 sp12 sp13 sp14 sp15
sp17 sp18 sp19 sp2 sp20 sp21 sp22
clump1 1 0 0 0 0 0 0 1 1 0 1 0 0 0
clump2a 0 1 0 0 0 0 0 1 0 0 1 0 0 1
clump2b 1 1 0 1 0 1 0 0 0 0 1 0 0 0
clump4 1 0 1 0 0 0 0 1 0 1 1 0 1 0
even 1 0 0 0 0 0 0 1 1 0 0 1 0 0
random 1 0 0 1 1 0 1 0 0 0 1 0 0 0
        sp24 sp25 sp26 sp29 sp3 sp4 sp5
sp6 sp7 sp8 sp9
clump1 0 0 0 1 1 0 0 1 0 0 1
clump2a 0 1 0 0 1 1 1 0 0 0 0
clump2b 0 0 0 0 1 1 0 0 0 0 1
clump4 0 0 0 0 0 0 1 0 0 1 0
even 1 0 1 0 0 0 0 0 1 0 1
random 0 1 0 0 0 1 1 0 0 0 0
```

cpr_iter_sim()

cpr_iter_sim() is not required for the CANAPE workflow, but is rather a diagnostic function for use with randomization algorithms that work by exchanging cells of the community matrix such as swap (Gotelli and Entsminger 2003) or curveball (Strona et al. 2014). Each such exchange is known as an 'iteration'. It is generally impossible to know a priori how many iterations are needed to completely randomize a given matrix; this number depends on properties of the matrix and the randomization algorithm. Generally, larger matrices with more skew (overabundance of zeros) will require more iterations (Miklós and Podani 2004).

cpr_iter_sim() conducts successive exchanges and records the similarity between the original matrix and the randomized matrix at each iteration. The similarity values should initially decrease until an approximate minimum is

Table 1. Datasets in canaper.

Dataset	n sites	n species (community matrix)	n species (phylogeny)
acacia	3037	508	510
biod_example	127	31	31
phylocom	6	25	32

The acacia dataset (Mishler et al. 2014) includes two outgroup taxa in the phylogeny that are not in the community matrix. The phylocom dataset (Webb et al. 2008) also includes more taxa in the phylogeny than the community matrix.

reached; further iterations will only result in noise around this minimum. A number slightly larger than the smallest number of iterations needed to reach the approximate minimum value can then be used for randomizing the community. The following code demonstrates usage of cpr_iter_sim() with the *Acacia* dataset (Mishler et al. 2014) that comes with canaper. The *Acacia* dataset is relatively large (Table 1) and highly skewed (98.1% zeros).

```
set.seed(12345)

data(acacia)

# Conduct up to 100,000 iterations
(swaps),
#   recording similarity every 1000
iterations
iter_sim_res <- cpr_iter_sim(
 comm = acacia$comm,
 null_model = "curveball",
 n_iterations = 100000,
 thin = 1000
)

# Inspect the output
iter_sim_res

# A tibble: 100 × 2
 iteration similarity
 <int> <dbl>
 1 1000 0.992
 2 2000 0.986
 3 3000 0.982
 4 4000 0.979
 5 5000 0.976
 6 6000 0.974
 7 7000 0.972
 8 8000 0.971
 9 9000 0.970
10 10000 0.969
# … with 90 more rows
```

It is useful to plot the output to identify the minimum number of iterations needed. Code to generate a plot with the ggplot2 package and its output are shown in the online Supporting information. The plot shows that the community matrix becomes maximally randomized after about 40 000–50 000 iterations and that the original matrix and randomized matrix are about 96.5% similar.

cpr_rand_test()

The cpr_rand_test() function carries out calculation of observed values, generation of random communities, and calculation of summary statistics as described above in 'Analysis workflow'. The main arguments to this function are the input community and phylogeny, type of null model, and settings

for the null model. Any species that do not appear in both the community data frame and the phylogeny will be excluded from analysis. For a full list of null models to choose from, run ?vegan::commsim(). It should be noted that the type of null model, number of random communities, and number of swapping iterations performed per random community (for swapping algorithms) all may strongly affect results of cpr_rand_test() (or any metric that is based on comparison to a set of random communities; Gotelli 2001). While it is beyond the scope of this paper to provide a full discussion of null models in ecology, we have provided details about how to explore appropriate null model settings with canaper in the 'How many randomizations?' vignette (https://docs.ropensci.org/canaper/articles/how-many-rand.html).

The output is a data frame with communities as rows and summary statistics in columns. A large number of summary statistics, including all of those needed to calculate CANAPE, are produced. For a full explanation of all output columns, see Appendix S1 or run ?cpr_rand_test().

The following code demonstrates usage of cpr_rand_test(), using the same example dataset as above.

```
set.seed(12345)
rand_test_results <- cpr_rand_test(
  phylocom$comm, phylocom$phy, null_
model = "curveball")

# Print a subset of the output columns
  rand_test_results[, 1:9]
     pd_obs pd_rand_mean pd_rand_sd pd_
obs_z pd_obs_c_upper
clump1  0.3018868  0.4654717  0.03350120
-4.882956 0
clump2a 0.3207547  0.4707547  0.03244554
-4.623132 0
clump2b 0.3396226  0.4720755  0.03696370
-3.583321 0
clump4  0.4150943  0.4683019  0.03374984
-1.576527 3
even    0.5660377  0.4654717  0.03339369
3.011528 100
random  0.5094340  0.4598113  0.03594522
1.380507 89
 pd_obs_c_lower pd_obs_q pd_obs_p_upper
pd_obs_p_lower
clump1 100 100 0.00 1.00
clump2a 100 100 0.00 1.00
clump2b 100 100 0.00 1.00
clump4 90 100 0.03 0.90
even 0 100 1.00 0.00
random 5 100 0.89 0.05

cpr_classify_endem()
```

The cpr_classify_endem() function classifies endemism types for the output of cpr_rand_test() as described above in 'Analysis workflow'. The input is a data frame including the following columns calculated by cpr_rand_test(): pe_obs_p_upper (upper p-value comparing observed PE to random values), pe_alt_obs_p_upper (upper p-value comparing observed PE on alternate tree to random values), rpe_obs_p_upper (upper p-value for RPE), and rpe_obs_p_lower (lower p-value for RPE). The output is the same data frame, with the column endem_type appended. Values of endem_type include paleo (paleoendemic), neo (neoendemic), not significant, mixed (mixed endemism), and super (super-endemic).

The following code demonstrates usage of cpr_classify_endem() with the output from cpr_rand_test() (note that for this small example, not all possible types of endem_type are produced).

```
canape_results    <-    cpr_classify_
endem(rand_test_results)

# Look at the endem_type column
canape_results[, "endem_type", drop =
FALSE]
          endem_type
clump1 not significant
clump2a not significant
clump2b not significant
clump4 not significant
even super
random mixed
```

In addition to cpr_classify_endem(), canaper includes the cpr_classify_signif() function for assessing significance level of PD, RPD, PE and RPE. Both cpr_classify_endem() and cpr_classify_signif() take a data frame as their first argument and return a data frame as output, so they are 'pipe friendly', i.e. can be chained together using pipe operators (%>% or |>).

Parallel computing

Parallel computing is enabled with the future package, which has been designed to allow maximum flexibility in the parallel backend selected by the user (e.g. multiple cores on one machine, multiple remote machines, etc.). Parallelization is applied to the calculation of summary statistics for each random community, as there are potentially many random communities (typically > 100 for a robust analysis, though this depends on the dataset). To use parallel computing, no changes are needed for cpr_rand_test() etc. Rather, future is loaded (library("future")), then a parallel back-end is specified with plan(). The user is advised to consult the future website (https://future.futureverse.org/) for more information on specifying a parallel backend.

The following code demonstrates parallel computing.

```
library(future)
set.seed(12345)
```

```
# Specify two cores running in parallel
plan(multisession, workers = 2)

# Usage of cpr_rand_test exactly the
same as before
rand_test_results <- cpr_rand_test(
    phylocom$comm, phylocom$phy, null_
model = "curveball")
```

There will not be a noticeable decrease in computing time since this dataset is so small. However, parallelization can greatly decrease computing time for large datasets; more details are available in the 'Parallel computing' vignette (https://docs.ropensci.org/canaper/articles/parallel.html).

Datasets

canaper comes with three datasets for testing and demonstration (Table 1). Each is a list with two named elements, comm (community matrix, data frame) and phy (phylogenetic tree, list of class "phylo"). acacia includes data of Australian *Acacia* analyzed by Mishler et al. (2014). phylocom and biod_example are synthetic datasets compiled for testing software, and were originally published in Phylocom (Webb et al. 2008) and Biodiverse (Laffan et al. 2010), respectively.

Example: Australian *Acacia*

Analysis with **canaper**

To demonstrate usage with a real dataset, we reproduced the analysis of Mishler et al. (2014), who conducted CANAPE on Australian *Acacia* using Biodiverse. All canaper analyses were run with canaper ver. 1.0.0 in R ver. 4.2.1. We used the curveball randomization algorithm with 50,000 iterations, since cpr_iter_sim() indicated this was sufficient to randomize the matrix ('Major functions'). Once the randomization algorithm and its settings have been selected, CANAPE can be conducted with just two commands.

```
# Set a seed for reproducibility
set.seed(12345)

# 1. Run randomization test
acacia_rand_res <- cpr_rand_test(
 acacia$comm, acacia$phy,
 null_model = "curveball",
 n_reps = 999, n_iterations = 50000,
 tbl_out = TRUE
)
# 2. Classify endemism
acacia_canape
<- cpr_classify_endem(acacia_rand_res)
```

canaper does not include any plotting functions to visualize the results. Rather, we recommend the ggplot2 package or base R graphics to visualize results. Here, we demonstrate use of the ggplot2 and patchwork packages to visualize the output of canaper (Fig. 2).

```
library(patchwork) # for multipart-panels
library(tidyverse) # includes ggplot2,
other packages for data tidying

# Fist do some data wrangling to make
the results easier to plot
# (add lat/long columns)
acacia_canape <- acacia_canape |>
 separate(site, c("long", "lat"), sep =
":") |>
      mutate(across(c(long,       lat),
parse_number))
min_long <- min(acacia_canape$long)
min_lat <- min(acacia_canape$lat)

a <- ggplot(acacia_canape, aes(x = long,
y = lat, fill = endem_type)) +
 geom_tile() +
 # Use same colors as Misher et al. 2014
# (for CVD-safe colors see cpr_endem_cols)
  scale_fill_manual(values    = mishler_
endem_cols) +
 guides(
  fill = guide_legend(title.position =
"top", label.position = "bottom")
 ) +
 coord_fixed() +
 # add a 1000 km scalebar (data units
are in m)
 geom_segment(
 x = min_long,
 y = min_lat,
 xend = min_long + 1000 * 1000,
 yend = min_lat
 ) +
 geom_text(
 x = min_long,
 y = min_lat,
 label = "1,000 km",
 vjust = -1, hjust = 0, size = 2
 ) +

 theme(
 legend.position = "bottom",
 legend.title = element_blank(),
 panel.grid.minor = element_blank(),
 axis.title = element_blank(),
 axis.text = element_blank(),
 axis.ticks = element_blank()
 )

b <- ggplot(
```
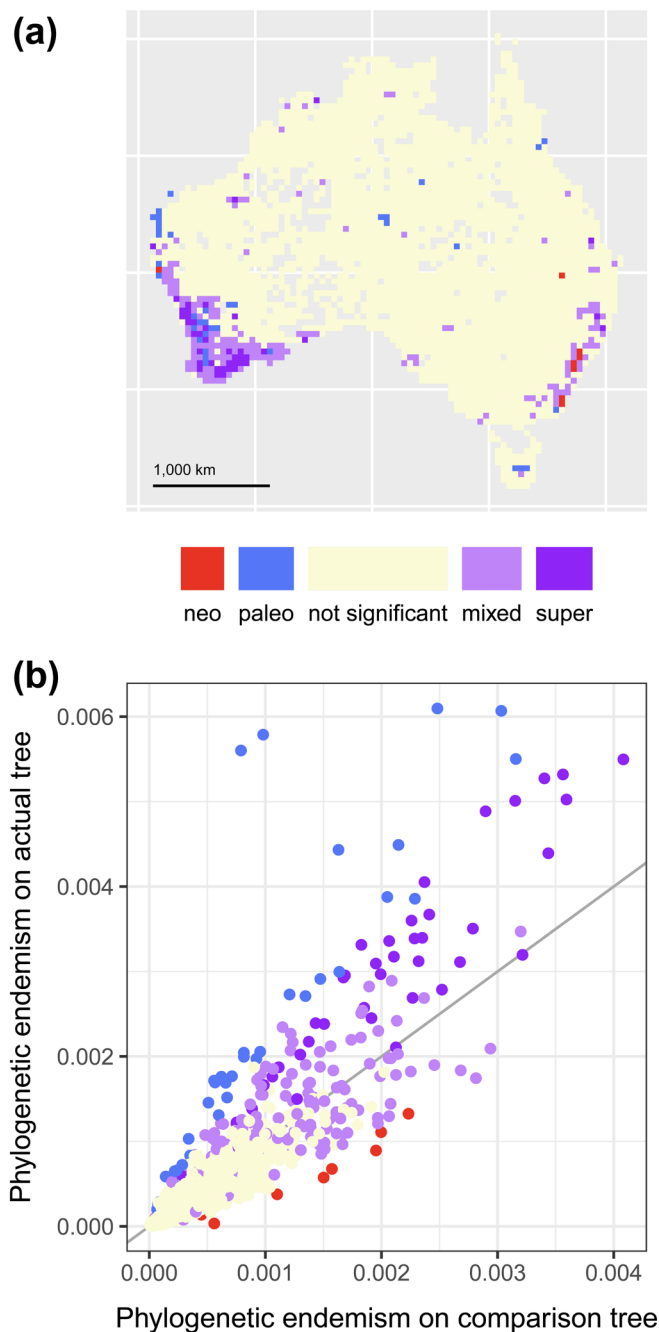
(a)


```
acacia_canape, aes(x = pe_alt_obs, y =
pe_obs, color = endem_type)) +
 geom_abline(slope = 1, color = "dark-
grey") +
 geom_point() +
 scale_color_manual(values = mishler_
endem_cols) +
 labs(
 x = "Phylogenetic endemism on compari-
son tree",
 y = "Phylogenetic endemism on actual
tree"
 ) +
 theme_bw() +
 theme(legend.position = "none")
a + b + plot_layout(ncol = 1) +
 plot_annotation(
 tag_levels = "a",
 tag_prefix = "(",
 tag_suffix = ")")
```

As in Fig. 3 of Mishler et al. (2014), grid-cells with significant endemism are primarily located on the coasts, with mostly non-significant grid-cells in the interior (Fig. 2). Furthermore, the endemism types largely correspond between the two figures.

The color palette used in Fig. 2, `mishler_endem_cols`, is the same as that used in Mishler et al. (2014). However, this palette may not be distinguishable to people with color-vision deficiency (CVD). Alternate color palettes (e.g. `cpr_endem_cols`) are also available in `canaper` that include CVD-safe colors (Okabe and Ito 2002). Palettes for plotting CANAPE and p-rank results can be selected with the `cpr_make_pal()` function. The same plot visualized with CVD-safe colors is available in the Supporting information.

(b)

Figure 2. Categorical analysis of neo- and paleo-endemism (CANAPE) of Australian *Acacia*. (a) Map of Australia showing grid-cells (communities) colored by endemism type. Latitude and longitude projected into equal area Australian Albers (EPSG:3577) coordinate system (Butler et al. 2007). (b) Scatterplot comparing phylogenetic endemism (PE) of each community as measured on the original tree vs. a comparison tree with all non-zero branch lengths set to equal length, colored according to endemism type. This figure reproduces Fig. 3 of Mishler et al. (2014).

Comparison with Biodiverse

We also re-ran the *Acacia* analysis using Biodiverse ver. 3.1 with the same settings as Mishler et al. (2014) (999 replicates of the `rand_structured` null model) and compared the results with those from `canaper`. Importantly, there is no expectation that results between the two should match exactly, for two reasons. First, the null model used between the Biodiverse and `canaper` analyses differ (`rand_structured` and `curveball`, respectively). `rand_structured` is not currently available in R, but we hope to add this to a future version of `canaper`. Second, the random communities generated in each run will be different, so the exact p-values will also be different. With a sufficiently high number of random communities, significance (e.g. at the is $\alpha = 0.5$ level) is expected to converge, but there may be borderline cases that appear significant in some analyses and non-significant in others. Note that we have verified that calculation of deterministic metrics that do not rely on random values, e.g. raw PD, are identical

Table 2. Comparison of CANAPE results for Australian *Acacia* between `canaper` and biodiverse.

Endemism type	n sites (`canaper`)	n sites (Biodiverse)
Mixed	172	170
Neo	8	7
Not significant	2780	2784
Paleo	36	35
Super	41	41

between Biodiverse and `canaper`, and include a test for this in `canaper`.

When we compared endemism type between the `canaper` and Biodiverse CANAPE results for Australian *Acacia*, they agreed in 98.4% of grid-cells (Table 2). The total number of significant cells was very similar (n = 257 and 253 sites, `canaper` and Biodiverse, respectively). The `rand_structured` and `curveball` null models both preserve total abundance per species and richness of each site (community matrix marginal sums); in that sense, they are relatively conservative null models (Strona et al. 2018), which may explain the relative high agreement between results. Selection of an appropriate null model is beyond the scope of this paper, but must be considered carefully in any community ecological analysis.

Computations were carried out on an MacBook Pro (2019) with 16 GB RAM and a 1.7 GHz, four core processor. Approximate compute times were 44.4 min for Biodiverse and 8.6 min for `canaper` (sequential, i.e. non-parallel mode, for both). However, when parallel computing was enabled for `canaper` with two cores, compute time dropped to 4.9 min. This demonstrates that `canaper` can efficiently compute moderately sized datasets with a personal laptop computer.

Comparison with other R packages

We are not aware of any other R packages that conduct the entire CANAPE pipeline automatically. However, there is a large number of packages for analyzing species diversity (approximately 40 packages out of 15 300 as of 2019; Pavoine 2020). Some of those that are more closely related to `canaper` include the following. `picante` (Kembel et al. 2010) was one of the first packages to offer calculation of phylogenetic alpha diversity such as MPD, MNTD, and PD (Webb 2000) and supports seven randomization algorithms, including two variations of the `swap` algorithm (Gotelli and Entsminger 2003, Miklós and Podani 2004). `phyloregion` (Daru et al. 2020) implements sparse matrix encoding to increase computing efficiency of PD and PE, and is used by `canaper`. `vegan` (Oksanen et al. 2017) performs a wide range of non-phylogenetic community ecology analyses. `vegan` includes by far the greatest number of algorithms for generating random communities, including `curveball` (Strona et al. 2014) and both variations of `swap`, and is used by `canaper`. `adiv` (Pavoine 2020) provides a flexible framework for analyzing alpha and beta diversity of biological communities based on either phylogenetic or trait distances.

`PhyloMeasures` (Tsirogiannis and Sandel 2016) implements efficient routines for calculating phylogenetic diversity metrics. `EcoSimR` (Gotelli and Ellison 2013) provides randomization algorithms as well as functions for characterizing community matrices and null distributions. As of writing, `PhyloMeasures` and `EcoSimR` had been removed from CRAN and may not be under active development.

Conclusions

The `canaper` package enables CANAPE completely within R for the first time. This will simplify workflows and facilitate reproducibility for researchers studying spatial biodiversity with R. Furthermore, `canaper` features efficient computing routines and simple yet flexible implementation of parallel computing, thereby decreasing computation time. `canaper` has already been used at least five studies while in development (Ellepola et al. 2022, Lu et al. 2022, Nitta et al. 2022, Naranjo et al. 2023, van Galen et al. 2023). We expect `canaper` will become a major tool in the toolkit of the emerging field of spatial phylogenetics alongside Biodiverse.

To cite `canaper` or acknowledge its use, cite this Software note as follows, substituting the version of the application that you used for 'ver. 1.0':

Nitta, J. H., et al. 2023. `canaper`: Categorical analysis of neo- and paleo-endemism in R. – Ecography 2023: e06638 (ver. 1.0).

Author contributions

Joel H. Nitta: Conceptualization (lead); Formal analysis (lead); Funding acquisition (supporting); Methodology (lead); Software (lead); Visualization (lead); Writing – original draft (lead); Writing – review and editing (equal). **Shawn W. Laffan**: Data curation (lead); Methodology (supporting); Software (supporting); Writing – review and editing (equal). **Brent D. Mishler**: Writing – review and editing (equal). **Wataru Iwasaki**: Funding acquisition (lead); Project administration (lead); Supervision (lead); Writing – review and editing (equal).

Transparent peer review

The peer review history for this article is available at https://publons.com/publon/10.1111/ecog.06638.

Data availability statement

The canaper R package is distributed by the Comprehensive R Archive Network (CRAN), with source code available on GitHub (https://github.com/ropensci/canaper) and Zenodo

(https://doi.org/10.5281/zenodo.5094032). Usage of canaper is documented at https://docs.ropensci.org/canaper/.

Code and data used to conduct analyses and generate this manuscript are available at https://github.com/joelnitta/canaper_ms, and can be run using the Docker image joelnitta/canaper_ms available at https://hub.docker.com/r/joelnitta/canaper_ms.

The acacia dataset was originally published by Mishler et al. (2014).

Supporting information

The Supporting information associated with this article is available with the online version.

References

Boettiger, C., Chamberlain, S., Hart, E. and Ram, K. 2015. Building software, building community: lessons from the rOpenSci project. – J. Open Res. Softw. 3: 8.

Butler, H., Schmidt, C., Springmeyer, D. and Livni, J. 2007. GDA94 / Australian Albers: EPSG projection. – https://spatialreference.org/ref/epsg/3577/.

Daru, B. H., Karunarathne, P. and Schliep, K. 2020. phyloregion: R package for biogeographical regionalization and macroecology. – Methods Ecol. Evol. 11: 1483–1491.

Diamond, J. 1975. Assembly of species communities. – In: Cody, M. L. and Diamond, J. (eds), Ecology and evolution of communities. Harvard University Press, pp. 342–444.

Ellepola, G., Pie, M. R., Pethiyagoda, R., Hanken, J. and Meegaskumbura, M. 2022. The role of climate and islands in species diversification and reproductive-mode evolution of Old World tree frogs. – Commun. Biol. 5: 347.

Faith, D. P. 1992. Conservation evaluation and phylogenetic diversity. – Biol. Conserv. 61: 1–10.

Gotelli, N. J. 2001. Research frontiers in null model analysis. – Global Ecol. Biogeogr. 10: 337–343.

Gotelli, N. J. and Entsminger, G. L. 2003. Swap algorithms in null model analysis. – Ecology 84: 532–535.

Gotelli, N. J. and Ellison, A. M. 2013. EcoSimR: null models for ecology. – https://www.uvm.edu/~ngotelli/EcoSim/EcoSim.html.

Kembel, S. W., Cowan, P. D., Helmus, M. R., Cornwell, W. K., Morlon, H., Ackerly, D. D., Blomberg, S. P. and Webb, C. O. 2010. Picante: R tools for integrating phylogenies and ecology. – Bioinformatics 26: 1463–1464.

Laffan, S. W., Lubarsky, E. and Rosauer, D. F. 2010. Biodiverse, a tool for the spatial analysis of biological and related diversity. – Ecography 33: 643–647.

Lai, J., Lortie, C. J., Muenchen, R. A., Yang, J. and Ma, K. 2019. Evaluating the popularity of R in ecology. – Ecosphere 10: e02567.

Lu, L., Zhao, L., Hu, H., Liu, B., Yang, Y., You, Y., Peng, D., Barrett, R. L. and Chen, Z. 2022. A comprehensive evaluation of flowering plant diversity and conservation priority for national park planning in China. – Fundam. Res. (in press).

Miklós, I. and Podani, J. 2004. Randomization of presence–absence matrices: comments and new algorithms. – Ecology 85: 86–92.

Mishler, B. D., Knerr, N., González-Orozco, C. E., Thornhill, A. H., Laffan, S. W. and Miller, J. T. 2014. Phylogenetic measures of biodiversity and neo- and paleo-endemism in Australian *Acacia*. – Nat. Commun. 5: 4473.

Mishler, B. D., Guralnick, R., Soltis, P. S., Smith, S. A., Soltis, D. E., Barve, N., Allen, J. M. and Laffan, S. W. 2020. Spatial phylogenetics of the North American flora. – J. Syst. Evol. 58: 393–405.

Naranjo, A. A., Folk, R. A., Gitzendanner, M. A., Soltis, D. E. and Soltis, P. S. 2023. Ancestral area analyses reveal Pleistocene-influenced evolution in a clade of coastal plain endemic plants. – J. Biogeogr. 50: 393–405.

Nitta, J. H., Mishler, B. D., Iwasaki, W. and Ebihara, A. 2022. Spatial phylogenetics of Japanese ferns: patterns, processes, and implications for conservation. – Am. J. Bot. 109: 727–745.

Okabe, M. and Ito, K. 2002. Color Universal Design (CUD) – How to make figures and presentations that are friendly to colorblind people. – https://jfly.uni-koeln.de/color/.

Oksanen, J., Blanchet, F. G., Friendly, M., Kindt, R., Legendre, P., McGlinn, D., Minchin, P. R., O'Hara, R. B., Simpson, G. L., Solymos, P., Stevens, M. H., Szoecs, E. and Wagner, H. 2017. vegan: Community ecology package ver. 2.4-4. – http://cran.r-project.org/package=vegan.

Pavoine, S. 2020. adiv: an R package to analyse biodiversity in ecology. – Methods Ecol. Evol. 11: 1106–1112.

Rosauer, D., Laffan, S. W., Crisp, M. D., Donnellan, S. C. and Cook, L. G. 2009. Phylogenetic endemism: a new approach for identifying geographical concentrations of evolutionary history. – Mol. Ecol. 18: 4061–4072.

Strona, G., Nappo, D., Boccacci, F., Fattorini, S. and San-Miguel-Ayanz, J. 2014. A fast and unbiased procedure to randomize ecological binary matrices with fixed row and column totals. – Nat. Commun. 5: 4114.

Strona, G., Ulrich, W. and Gotelli, N. J. 2018. Bi-dimensional null model analysis of presence-absence binary matrices. – Ecology 99: 103–115.

Thornhill, A. H., Baldwin, B. G., Freyman, W. A., Nosratinia, S., Kling, M. M., Morueta-Holme, N., Madsen, T. P., Ackerly, D. D. and Mishler, B. D. 2017. Spatial phylogenetics of the native California flora. – BMC Biol. 15: 96.

Tsirogiannis, C. and Sandel, B. 2016. PhyloMeasures: a package for computing phylogenetic biodiversity measures and their statistical moments. – Ecography 39: 709–714.

van Galen, L. G., Orlovich, D. A., Lord, J. M., Nilsen, A. R., Dutoit, L. and Larcombe, M. J. 2023. Correlated evolution in an ectomycorrhizal host–symbiont system. – New Phytol. 238: 1215–1229.

Webb, C. O. 2000. Exploring the phylogenetic structure of ecological communities: an example for rain forest trees. – Am. Nat. 156: 145–155.

Webb, C. O., Ackerly, D. D. and Kembel, S. W. 2008. Phylocom: software for the analysis of phylogenetic community structure and trait evolution. – Bioinformatics 24: 2098–2100.