BIODIVERSITY BUILDING BLOCKS FOR POLICY

D5.2 Phylogenetic Diversity

30/11/2024

Author(s):
Lissa Breugelmans, Louise Antonia Hendrickx, Quentin Groom, Joe Miller, Ward Langeraert, Shawn Dove, Hanno Seebens, Michele Di Musciano, Duccio Rocchini, Maarten Trekels

Funded by the European Union

Prepared under contract from the European Commission
Grant agreement No. 101059592
EU Horizon Europe Research and Innovation Action

Project acronym:	**B3**
Project full title:	**Biodiversity Building Blocks for policy**
Project duration:	01.03.2023 – 31.08.2026 (42 months)
Project coordinator:	Dr. Quentin Groom, Agentschap Plantentuin Meise (Meise BG)
Call:	HORIZON-CL6-2021-GOVERNANCE-01
Deliverable title:	[Phylogenetic Diversity]
Deliverable n°:	[D5.2]
WP responsible:	[WP5]
Nature of the deliverable:	[Report]
Dissemination level:	[Public]
Licence of use:	Creative Commons Attribution 4.0 International
Lead partner:	[MBG]
Recommended citation:	Breugelmans, L., Hendrickx, L. A., Groom, Q., Miller, J., Langeraert, W., Musciano, M. D., Dove, S., Seebens, H., Rocchini, D., Trekels, M. (2024). ***Phylogenetic Diversity***. B3 project deliverable D5.2.
Due date of deliverable:	Month n° 21
Actual submission date:	Month n° 21

Deliverable status:

Version	Status	Date	Author(s)
1.0	Final	29 November 2024	Lissa Breugelmans (MeiseBG), Louise Antonia Hendrickx (MeiseBG), Quentin Groom (MeiseBG), Joe Miller (GBIFS), Ward Langeraert (EV_INBO), Michele Di Musciano (UNIVAQ), Shawn Dove (JLU), Hanno Seebens (JLU), Duccio Rocchini (UNIBO), Maarten Trekels (MeiseBG)

Table of contents

Key takeaway messages

- We developed and released the R-package `pdindicatoR` for producing phylogenetic diversity maps and calculating a phylogenetic diversity indicator based on species occurrence cubes and phylogenetic trees
- This package includes functions to match tree tip labels with the Open Tree Taxonomy (OTT) and with the species in the occurrence cube based on the GBIF taxonKey, calculate phylogenetic diversity metrics, generate PD maps and calculate a PD indicator.
- Documentation is available as a README file and vignettes, which detail an example workflow and customization options. The documentation is also provided on the B3 tutorial website.
- The tool will be tested on the case studies of WP6. User feedback is captured in the issue tracker on GitHub and will be implemented as far as feasible during the remainder of the project.

Executive summary

We developed and released the R-package `pdindicatoR` to produce phylogenetic diversity maps and calculate a phylogenetic diversity indicator based on species occurrence cubes and phylogenetic trees. The produced maps show the spatial distribution of phylogenetic diversity as well as the location of protected areas, and they can for example be used to inform policymakers and as a planning tool for conservation managers. The software includes functions to match tree tip labels with the Open Tree Taxonomy (OTT) and with the species in the occurrence cube based on the GBIF taxonKey, calculate phylogenetic diversity metrics, generate PD maps, and calculate a PD indicator. Documentation and example workflows are provided. The tool will be tested on the case studies of WP6 and further refined based on user feedback.

Non-technical summary

Phylogenetic diversity (PD) is a measure of biodiversity that not only takes the number of different species into account, but also their evolutionary history (relatedness). This is important because by conserving phylogenetic diversity globally, we conserve the full variety of different evolutionary features of species and thus future options for humanity. Biodiversity indicators are tools for evaluating changes in living organisms and ecosystems. We developed a software tool called `pdindicatoR that` can produce maps of phylogenetic diversity and calculate how much of PD is safeguarded by protected areas. It can be used as a tool to locate hotspots of phylogenetic diversity, and can aid in making informed decisions on where to expand existing protected areas or create new ones.

List of abbreviations

API	Application Programming Interface
B3	B-cubed project
EBV	Essential Biodiversity Variable
EDGE	Evolutionarily Distinct & Globally Endangered
EEA	European Environment Agency
EU	European Union
GBIF	Global Biodiversity Information Facility
IPBES	Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services
IUCN	International Union for Conservation of Nature
MRCA	Most Recent Common Ancestor
OTL	Open Tree of Life
OTT	Open Tree of Life Taxonomy
PD	Phylogenetic Diversity

1. Introduction

The world is changing rapidly; climate change, land use change, pollution and natural resource exploitation are creating a global crisis for biodiversity whose magnitude and dynamics are hard to quantify. Decision-makers at all levels need up-to-date information from which to evaluate policy options. For this reason rapid, reliable, repeatable monitoring of biodiversity data is needed at all scales from local to global. The overarching goal of B3 is to provide easy access to tools in a cloud computing environment, in real time and on-demand, with state-of-the-art prediction models of biodiversity, that will output models and indicators of biodiversity status and change.

In WP5, we want to create reproducible and sustainable workflows to calculate indicators and their uncertainty based on aggregated occurrences, as well as potential species distribution resulting from spatio-temporal interpolation or niche modelling. One of these indicators is the Phylogenetic Diversity indicator.

1.1 Phylogenetic Diversity

Phylogenetic diversity (PD) is a measure of biodiversity that takes evolutionary history into account by estimating how evolutionary related or distinct species are [1]. PD has many possible applications; it can be used as a guide for conservation purposes, as is the case for the EDGE of Existence Programme, or as a general measure for biodiversity in which we should not aim to conserve specific features, but rather to maximise a variety of features [2,3]. The latter could be particularly useful in light of the changing environmental conditions, as we can only guess which features will be important in the future.

Many measures of biodiversity exist and they all have their value depending on what you want to calculate or investigate. The most intuitive one is probably species richness, which is defined as the number of species within a given region [4]. It is easy to understand and to measure and can often be used as a surrogate for other measures that are harder to quantify [5]. Species richness, however, does not take each individual species' identity into account. It considers all species as equally valuable [3], which is more nuanced in reality. It is also more sensitive to taxonomic inflation associated with sampling effort [6].

Ecosystem functioning is influenced by more than just the number of species, making measures like functional trait diversity and phylogenetic diversity particularly important. Functional traits, which describe how species acquire, share, and conserve resources, are fundamental to understanding their roles in ecosystem processes. Violle et al. (2007) define functional traits as "*morpho-physio-phenological traits which impact fitness indirectly via their effects on growth, reproduction, and survival—the three components of individual performance*" [8]. These traits underpin both contributions to ecosystem properties and services, as well as species' tolerance to environmental stressors and disturbances [9].

Phylogenetic diversity offers a practical alternative for assessing biodiversity. It is often more feasible to calculate than functional diversity and has been shown to predict biodiversity effects

similarly [10]. Furthermore, there is increasing evidence that ecologically relevant traits often exhibit phylogenetic conservatism, meaning that phylogenetic diversity can capture important functional aspects of diversity [11–13]. This makes phylogenetic diversity a valuable tool for studying ecosystem functioning, particularly when direct functional trait data are unavailable.

That said, users should be aware of the contexts where phylogenetic diversity and functional diversity may diverge. For example, convergent evolution can result in species with high phylogenetic diversity but similar traits, leading to lower functional diversity. Conversely, closely related species that have diversified functionally can result in low phylogenetic diversity but high functional diversity. While these cases exist, they do not diminish the overall utility of phylogenetic diversity as an effective and broadly applicable measure for biodiversity studies, provided its limitations are considered [10].

Some studies suggest that PD would promote ecosystem stability, most likely through reduced competition or increased facilitative interactions for communities with distantly related species [15,16]. It has therefore been suggested to maximise evolutionary diversity if the goal is to maximise community functionality in habitat restoration [17]. On the other hand, a re-analysis of 16 PD experiments conducted in grassland ecosystems has questioned these findings. By using an updated phylogeny, they found that species richness was a superior predictor of community productivity and stability compared to PD. Although they found a significant relationship between PD and community productivity, it was not sufficiently related to ecosystem stability and was therefore concluded to be an inferior predictor compared to species richness [18]. While PD remains a valuable metric for understanding community diversity and its potential role in driving ecosystem processes, these findings highlight the need for further research to determine whether it consistently outperforms other predictors, such as species richness or functional trait diversity, in explaining community productivity and stability; automated workflows to study PD across different taxonomic groups, habitats, and scales could help address some of these open questions.

Attempts to reduce biodiversity loss are not helped by the existence of numerous measures of biodiversity, combined with a lack of consensus about which one to monitor. The Essential Biodiversity Variables (EBV) framework tries to tackle this by defining a minimum set of essential measurements to capture the major dimensions of biodiversity change [19]. The difficulty of having 6 EBV classes with 21 EBV names and respective metrics is that in a real ecosystem, obtaining measurements of even this minimum set will be difficult. According to *Lean and Maclaurin* (2016), a more general measure for biodiversity is needed for large-scale environmental decision-making [3]. They suggest conserving a maximal variety of features instead of conserving particular features. Since it is difficult to measure features directly, they suggest using PD as a general measure, as it reflects the evolutionary processes that cause functional & morphological divergence within lineages.

Within the maze of biodiversity measures, a multitude of PD metrics adds even more complexity. Luckily, *Tucker et al.* (2017) provide some clear guidelines for making informed decisions about the use of PD metrics. They collected 70 existing PD metrics and divided them

into three dimensions: richness, divergence, and regularity [20]. Faith's PD index, for example, is one of the most widely used metrics of phylogenetic diversity and is defined as the sum of the length of phylogenetic tree branches representing the minimum tree-spanning path among a group of species. According to the mentioned guidelines, this metric would be categorised in the richness dimension and can be used when you want to calculate what the total evolutionary history is within (or between) assemblages.

1.2 Policy relevance

Slowly but surely, phylogenetic indicators are being incorporated into policy. First, some IPBES assessments have used a PD indicator as the percentage of a taxonomic group's PD that is represented by threatened species and is recognised by IPBES as an indicator for 'maintenance of options' [21, 22]. Second, the expected loss of PD is an indicator recognised by the Global Biodiversity Framework and represents the amount of evolutionary history expected to be lost in a given amount of time based on the current extinction risks faced by the set of species [23]. Third, EDGE provides a priority list of species that are evolutionarily distinct and globally endangered [23]. Finally, a PD Task Force exists within the framework of the IUCN, who provide leadership, guidance and expertise on the inclusion of PD in conservation strategies for practitioners, decision-makers, and the public.

These three indicators most currently used in policy, are all species-based rather than area-based. An indicator that gives information about how well PD of a certain higher taxonomic group is currently safeguarded by protected areas and a spatial visualisation which can be used to identify potential directions for future expansion of protected areas is thus of particular interest. Therefore, we designed a workflow (see *Figure 1*) to calculate and visualize the overlap between protected areas and highly phylogenetically diverse areas. The output of this workflow are phylogenetic diversity distribution maps for the area and taxonomic group of interest, and an indicator value that represents the percentage of high phylogenetically diverse grid cells that fall within protected areas. The PD distribution maps can be used by nature management professionals to identify potential directions for future expansion of protected areas. The indicator value represents how well the phylogenetic diversity within a taxon of interest is currently safeguarded within protected areas. The indicator can be calculated for different geographical scales by providing the appropriate species occurrence cube. It can also be used to evaluate the impact of designating new protected areas on safeguarded phylogenetic diversity for a taxon of interest.

1.3 Landscape analysis

Earlier work by Vladimir Mikryukov resulted in the development of `PhyloNext` [24], an automated pipeline to analyse phylogenetic diversity using GBIF occurrence data, species phylogenies from Open Tree of Life (OTL) [25], and `Biodiverse` [26]. Built using `Nextflow`

[27], it uses a separate `Docker` [28] container for each process, making it easier to maintain software dependencies and making the results highly reproducible. The pipeline could also be launched in a cloud environment. Biodiverse can generate more than 300 different biodiversity metrics, including a wide range of phylogenetic indices.

Because of the reliability and versatility of `PhyloNext`, we initially set out to adapt the pipeline to take data cubes as input, in order to produce a workflow for producing the phylogenetic indicator within the B3 project. However, we encountered a number of challenges:

- `Docker` containers might be reliable and portable, but the objective of B3 is to integrate the produced indicators in one readily available 'package'. It is more straight-forward to integrate all indicators in an overarching metapackage when developed as individual `R`-packages.
- `BiodiverseR` [29], the `R` interface to `Biodiverse`, is currently under development. Because Biodiverse is largely coded in `Perl`, running it as stand-alone software (outside of a `Docker` container) resulted in a lot of dependency issues.
- A large part of the `Phylonext` workflow consists of data cleaning, which is already covered at the occurrence data cube creation step.

For these reasons, it was decided to develop an `R`-package that contains functions to combine occurrence data cube and phylogenetic tree data and calculate PD metrics from scratch.

2. Methods

2.1 Software description

We released an R-package called `pdindicatoR` [30] which is the implementation of the specifications outlined in *Milestone 22 Design of the phylogenetic indicators*. The source code is freely available on GitHub (https://github.com/b-cubed-eu/pdindicatoR) and the package can be installed from this repository in the following manner:

```
Unset
# install.packages("remotes")
remotes::install_github("b-cubed-eu/pdindicatoR")
```

Releases are automatically deposited on Zenodo (10.5281/zenodo.14237551).

The package contains functions to process a species occurrence data cube and a phylogenetic tree, calculate phylogenetic diversity metrics based on this input, generate a gridded PD map for the focus-area and calculate the percentage of high PD grid cells safeguarded within protected areas.

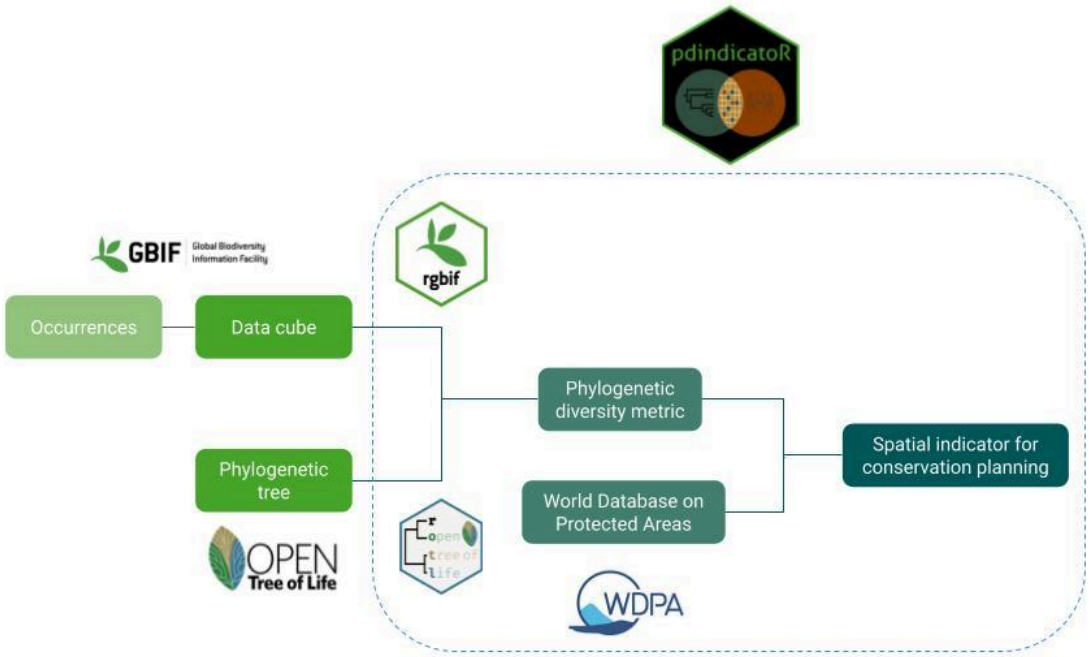


Figure 1: The pdIndicatoR workflow. By Lissa Breugelmans, licensed under CC BY 4.0

The following functions are provided:

- **taxonmatch()**
- **append_ott_id()**
- **check_completeness()**
- **aggregate_cube()**
- **get_pd()**
- **calculate_faithpd()**
- **convert_multipolygons()**
- **generate_map_and_indicator()**
- **make_shiny_maps()**

taxonmatch()

This function matches the tip labels of a phylogenetic tree (taxon names or OTT id's) with their corresponding GBIF id's. It uses the function `rotl::tnrs_match_names()` to call on OTL's taxonomic name resolution service API, which matches taxonomic names with the OTL taxonomic backbone and returns the corresponding Open Tree of Life Taxonomy (OTT) id and linked GBIF taxonkey.

Unset

```
#' Taxon matching
#'
#' @param tree An object of class 'phylo', iow a phylogenetic tree in Newick or
#' Nexus format that was parsed by ape::read_tree()
#' @result A dataframe with columns ott_id and gbif_id
#' @example mtable <- taxonmatch(tree_musceloidea)
#' head(mtable)
#' @export

taxonmatch <- function(tree) {
  tree_labels <- tree$tip.label

  if (any(stringr::str_detect(tree_labels,'ott\\d+')) == FALSE){
    taxa <- rotl::tnrs_match_names(tree_labels)
  } else {
    taxa <- data.frame(tree_labels)
    colnames(taxa)[1] <- "ott_id"
  }

  taxa[,"gbif_id"] <- NA
  i=1
  for(id in taxa$ott_id){
    if (is.na(id) == FALSE){
      tax_info <- rotl::taxonomy_taxon_info(id)
      for(source in tax_info[[1]]$tax_sources){
        if (grepl('gbif', source, fixed=TRUE)){
          gbif <- stringr::str_split(source,":")[[1]][2]
          taxa[i,]$gbif_id <- gbif
        }}}
    i = i + 1}
  taxa$gbif_id <- as.integer(taxa$gbif_id)

  original_df <- data.frame(
    orig_tiplabel = unique(tree_labels),
    search_string = tolower(unique(tree_labels)))

 matched_result <- merge(taxa, original_df, by = "search_string", all.x = TRUE)
 return(matched_result)}
```

append_ott_id()

This function uses the table produced by the `taxonmatch()` function to create a linking table, and then appends the ott_id's as a new variable to the provided occurrence cube.

```
Unset
#' Append ott id's to cube
#'
#' @param tree An object of class 'phylo', iow a phylogenetic tree in Newick
#' format that was parsed by ape::read_tree()
#' @param cube A dataframe with for selected taxa, the number of occurrences
#' per taxa and per grid cell
#' @param matched A dataframe, returned by running the function taxonmatch() on
#' a phylogenetic tree, which contains the tip labels of the tree and their
#' corresponding gbif_id's
#' @return A dataframe which consist of all the data in the original datacube,
#' appended with column ott_id
#' @example
#' @export

append_ott_id <- function(tree, cube, matched){

    # Append OTT id's to occurrence cube
  speciesKeys <- cube["specieskey"] %>% distinct()

  mtable <- speciesKeys %>% left_join(matched[,c("ott_id","gbif_id",
"unique_name", "orig_tiplabel")],
                                      by = join_by(specieskey == gbif_id))

  mcube <- cube %>% left_join(mtable[,c("specieskey", "ott_id", "unique_name",
"orig_tiplabel")],
                              by = join_by(specieskey == specieskey))
  return(mcube)
  }
```

check_completeness()

This function checks to which degree the provided phylogenetic tree covers the species in the occurrence cube. In this way, the user can check whether the selected tree contains enough information to proceed with the analysis, or if a new tree should be selected (species that can not be matched are removed from the dataset in further analysis steps).

```
Unset
#' Check if provided phylogenetic tree is complete and covers all species in
#' occurence cube
#'
#' @param mcube A dataframe which is returned by the function append_ott_id(),
#' and contains the occurence datacube with ott_id variable appended.
#' format that was parsed by ape::read_tree()
#' @return a list - first element is the total number of species in the
#' occurence cube, second element is the number of species lacking in the
#' phylogenetic tree.
#' @example
#' @export
check_completeness <- function(mcube){

  mcube_dist <- distinct(mcube, specieskey, .keep_all=TRUE)
  sp_na <- mcube_dist %>% dplyr::filter(is.na(ott_id)) %>% select(specieskey,
                                                                  species)
  # sp_miss <- paste(sp_na,collapse=",")
  cat("The following species are not part of the provided phylogenetic
tree:\n")
  print(sp_na)
}
```

aggregate_cube()

This function aggregates a provided datacube over grid cell id, and produces a new datacube with three variables (`speciesKeys`, `ott_id's`, `unique_names` and `orig_tiplabels`) that contain the lists of species that are observed for each grid cell. The function has an optional argument timegroup, which takes an integer x as input value. When this argument is provided, the occurrences are aggregated for fixed time periods of x years. An observed species list is then generated for each timegroup and grid cel combination. The result will be that a PD map and indicator will be generated for each time period.

```
Unset

#' Aggregate datacube over grid cell to create new dataframe with species list
per grid
#'
#' @param mcube An occurence datacube with appended ott_id's, as produced by
#' the append_ott_id function
#' @param cube A dataframe with for selected taxa, the number of occurrences
#' per taxa and per grid cell
#' @return A dataframe with for each grid cell
#' @example
#' @export
```

```
aggregate_cube <- function(mcube, timegroup=NULL) {
  columns_to_select <- c("year", "eeacellcode", "specieskey", "ott_id",
"unique_name", "orig_tiplabel")
  simpl_cube <- mcube[, intersect(columns_to_select, colnames(mcube))]
  min_year <- min(simpl_cube$year)

  # When occurrences are already aggregated over time or when no timegroup is
  # specified
  if (!("year" %in% colnames(simpl_cube)) || missing(timegroup)) {
    aggr_cube <- simpl_cube %>%
      group_by(eeacellcode) %>%
      reframe(
        specieskeys = list(unique(specieskey)),
        ott_ids = list(unique(ott_id)),
        unique_names = list(unique(unique_name)),
        orig_tiplabels = list(unique(orig_tiplabel))
      )

  # When timegroup ==1
  } else if(timegroup==1){
      aggr_cube <- simpl_cube %>% arrange(year) %>%
      group_by(eeacellcode, year) %>%
      reframe(
        specieskeys = list(unique(specieskey)),
        ott_ids = list(unique(ott_id)),
        unique_names = list(unique(unique_name)),
        orig_tiplabels = list(unique(orig_tiplabel))
      )%>%
      rename(period = year)
  } else {

  # Calculate the 5-year period for each row
   aggr_cube <- simpl_cube %>% arrange(year) %>%
    mutate(period = min_year + 5 * ((year - min_year) %/% 5)) %>%
     mutate(period = paste(period, period + 4, sep = "-")) %>%
    group_by(period, eeacellcode) %>%
     reframe(
       specieskeys = list(unique(specieskey)),
       ott_ids = list(unique(ott_id)),
       unique_names = list(unique(unique_name)),
       orig_tiplabels = list(unique(orig_tiplabel))
     )
  }
  return(aggr_cube)}
```

get_pd()

This function retrieves the tip labels of all the species in the matched datacube `mcube` and calculates their most recent common ancestor (MRCA) based on the provided phylogenetic tree. The argument metric can be used to specify which PD metric(s) need to be generated. If this argument is not provided, the default value is `faith` which will calculate Faith's PD. It is envisioned to extend the possible values for this argument, as new functions are created for the calculation of additional PD metrics.

Unset

```
#' Find MRCA for data cube and call function to calculate PD metrics
#'
#' @param tree A phylogenetic tree with branch lengths
#' @param species A character vector with species names
#' @return Calculated PD value
#' @example get_pd(tree, species)
#' @export
#'

get_pd <- function(tree, species, metric="faith"){

# get all species in matched cube

all_matched_sp<-unique(mcube[["orig_tiplabel"]])

# find most recent common ancestor
MRCA <- ape::getMRCA(tree, all_matched_sp)

# calculate PD metric
if (metric=="faith"){
calculate_faithpd(tree, species, MRCA)
}
}
```

calculate_faithpd()

This function calculates Faith's phylogenetic diversity, based on a provided list of species and a phylogenetic tree. Faith's PD was originally described by Faith in 1992 [2] and is calculated as the total phylogenetic branch length spanning all taxa of a community. The calculation should extend to the root of the tree (ancestral to all taxa considered in the study) [*Figure 2*, 31], such that a single-taxon community will not have PD = 0. Therefore, this function has the argument MRCA, which is the node number of all the species in the analysis (see function get_pd, which determines the MRCA of all species in the datacube).

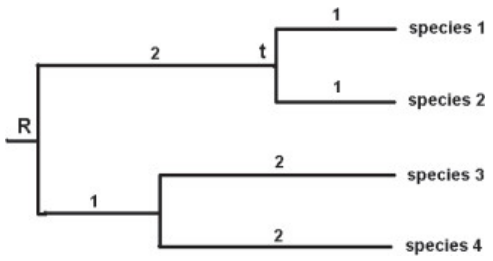


Figure 2: A phylogenetic tree representing the evolutionary relationships between four species under study. Faith's PD for species 1 and species 2 is calculated as the sum of the branch lengths to the common root of the tree (also known as the most recent common ancestor (MRCA) of the four species) and equals 4. Figure from Faith & Baker, 2006 [31] under CC by NC 3.0

Unset

```
#' Calculation of Faith's PD
#'
#' This function calculates Faith's PD, based on a provided list of species
#' and a phylogenetic tree.
#'
#' @param tree An object of class 'phylo', iow a phylogenetic tree in Newick
#' format that was parsed by ape::read_tree()
#' @param species A character vector where each element is a species, and more
#' specifically, matches a tip label of the phylogenetic tree exactly
#' @return A string that combines "Phylogenetic diversity:" and the calculated
#' value
#' @example
#' # Create random tree
#' rtree <- rtree(10, rooted=TRUE)
#' str(rtree)
#' rtree$tip.label
#' tree$node.label
#' tree$edge
#'
#' plot(rtree)
#' nodelabels()
#' edgelabels()
#' tiplabels()
#'
#' # Create vector with selected leaves/species for which to calculate the PD
#' trio <- c("t3", "t5", "t1")
#'
#' # Determine MRCA of all species under study. If the group of species under
#' # study contains of t1, t2, t3, t4, t5, t6
#' all_species <_ c(t1, t2, t3, t4, t5, t6)
#' MRCA <- get_MRCA(tree, all_species )
#'
#' # Calculate PD for trio of observed species
```

```
#' calculate_faithpd(rtree, trio, MRCA)
#' @export


calculate_faithpd <- function(tree, species, MRCA){

  # get tip id's from tip labels

   tip_ids <- vector(mode="integer", length=length(species))
   for (i in seq_along(species)) {
       x <- which(tree$tip.label == species[i])
       tip_ids[i] <- x
   }

  # determine spanning paths (nodes) from species to MRCA

  nodepath <- vector(mode="list", length(tip_ids))
  for (i in seq_along(tip_ids)){
    x <- ape::nodepath(tree, MRCA, tip_ids[i])
    nodepath[[i]] <- x
  }

  # get the branches/edges along the spanning paths

  edge_ids <- vector(mode="list", length(tip_ids))
  for (i in seq_along(tip_ids)){
    edges <- which(tree$edge[, 1] %in% nodepath[[i]][-length(nodepath[[i]])] &
                     tree$edge[, 2] %in% nodepath[[i]][-1])
    edge_ids[[i]] <- edges
  }

  # Count shared branches only once

  edge_ids_unique <- unique(unlist(edge_ids))

  # Sum the length of the branches

  edge_lengths <- tree$edge.length[edge_ids_unique]
  pd <- sum(edge_lengths)
  #print(paste("Phylogenetic diversity:", pd))
}
```

convert_multipolygons()

This function is a helper function to convert a multi-surface object, generated by joining an occurrence datacube with a grid shapefile, into a multipolygon object. It is called by the `generate_map_and_indicator()` function.

```
Unset
#' Convert multisurface object to multipolygon object
#'
#' @param object An object of class multisurface
#' @return An object of class multipolygon
#' @example convert_multipolygons(cube_highPD)
#' @export


convert_multipolygons <- function(object) {
  tmp1 <- tempfile(fileext = ".gpkg")
  tmp2 <- tempfile(fileext = ".gpkg")
  sf::st_write(object, tmp1)
  gdalUtilities::ogr2ogr(tmp1, tmp2, f = "GPKG", nlt = "MULTIPOLYGON")
  Y <- sf::st_read(tmp2)
  sf::st_sf(sf::st_drop_geometry(object), geom = sf::st_geometry(Y))
}
```

generate_map_and_indicator()

This function creates one or multiple map(s) visualizing the calculated PD metric(s) for each grid cell and the boundaries of protected areas. If the optional argument cut-off=x is provided, this function also calculates the percentage of high PD cells (as determined by the cut-off value) that currently fall within the boundaries of protected areas. The map(s) and/or indicator values are stored in the resulting list variable PDindicator. The optional argument bbox_custom can be provided to generate alternative mapviews (zooming in or out) and should be specified as a vector with the coordinates of the left bottom and right upper corner of the desired bounding box. The shapefile with the protected area polygons is provided by the user.
(e.g. Natura 2000 polygons from the World Database of Protected Areas).

```
Unset
#' Mapping PD and calculating indicator

#' @param PD_cube An sf dataframe containing the calculated PD metrics (column
name
#' 'PD') for each grid cell with occurrences of a selected higher taxon, and
the
```

```
#' geometries of those grid cells.
#' @param grid An sf object with variable detailing grid cell codes and a
#' geometry column
#' @param taxon A selected higher taxon, for which the occurrence cube was
#' generated. Used to generate the map's title only.
#' @param bbox_custom Optional, numeric vector with custom bounding box
#' coordinates as c(xmin, xmax, ymin, ymax)
#' @param cutoff A variable of type numeric which determines the cut-off point
#' between low PD and high PD
#' @return
#' @example
#' bbox <- c(3885477, 3929441,3103857, 3126672)
#' map_PD <- PD_map(PD_cube, grid, "Musceloidea", bbox_custom=bbox,
cut-off=150)
#' print(map_PD[[1]])

generate_map_and_indicator <- function(PD_cube, grid, taxon = NULL, bbox_custom
= NULL, cutoff = NULL) {

# Merge grid with cube
PD_cube_geo <- right_join(grid, PD_cube, by = join_by(CELLCODE == eeacellcode))

# Set bounding box
if (is.null(bbox_custom)) {
  bbox <- st_bbox(PD_cube_geo)
} else {
  if (length(bbox_custom) != 4) {
    stop("bbox_custom must be a numeric vector of length 4: c(xmin, xmax, ymin,
ymax).")
  }
  bbox <- c(xmin = bbox_custom[1], xmax = bbox_custom[2],
            ymin = bbox_custom[3], ymax = bbox_custom[4])
}

# Expand bounding box
expansion_factor <- 0.20
bbox_expanded <- c(
  xmin = as.numeric(bbox["xmin"]) - (as.numeric(bbox["xmax"]) -
as.numeric(bbox["xmin"])) * expansion_factor,
  xmax = as.numeric(bbox["xmax"]) + (as.numeric(bbox["xmax"]) -
as.numeric(bbox["xmin"])) * expansion_factor,
  ymin = as.numeric(bbox["ymin"]) - (as.numeric(bbox["ymax"]) -
as.numeric(bbox["ymin"])) * expansion_factor,
  ymax = as.numeric(bbox["ymax"]) + (as.numeric(bbox["ymax"]) -
as.numeric(bbox["ymin"])) * expansion_factor
)
```

```
# Read in country borders
world <- rnaturalearth::ne_countries(scale = "medium", returnclass = "sf")
world_3035 <- sf::st_transform(world, crs = 3035)

# Initialize lists to store maps and indicators
plots <- list()
indicators <- list()

# Calculate global min and max PD values for consistent color scale
pd_min <- min(PD_cube_geo$PD, na.rm = TRUE)
pd_max <- max(PD_cube_geo$PD, na.rm = TRUE)

# Check for 'period' column in PD_cube
if ("period" %in% colnames(PD_cube_geo)) {
  unique_periods <- unique(PD_cube_geo$period)

  for (p in unique_periods) {
    # Subset data for the current period
    current_period_data <- PD_cube_geo %>% filter(period == p)

    # Create the map for the current period
    map <- ggplot() +
      geom_sf(data = world_3035, fill = "antiquewhite") +
      geom_sf(data = current_period_data, mapping = aes(fill = PD)) +
      scale_fill_viridis_c(option = "B", limits = c(pd_min, pd_max)) +
      geom_sf(data = pa, fill = NA, color = "darkgreen", linewidth = 0.05) +
      coord_sf(xlim = c(bbox_expanded["xmin"], bbox_expanded["xmax"]),
               ylim = c(bbox_expanded["ymin"], bbox_expanded["ymax"]), expand =
FALSE) +
      xlab("Longitude") + ylab("Latitude") +
      labs(title = paste("Taxon:", taxon),
           subtitle = paste("Phylogenetic Diversity for period:", p)) +
      theme(panel.grid.major = element_line(color = gray(0.5), linewidth =
0.5),
            panel.background = element_rect(fill = "aliceblue"))

    # Store the plot in the list
    plots[[as.character(p)]] <- map


    # Calculate indicator for the current period if cutoff is provided
    if (!is.null(cutoff)) {
      current_period_data$PD_high <- ifelse((current_period_data$PD > cutoff),
1, 0)
```

```
      cube_highPD <- current_period_data[current_period_data$PD_high == 1,
c("OBJECTID", "CELLCODE", "PD", "geom", "PD_high")]

      # Convert to multipolygon object
      cube_mp <- convert_multipolygons(cube_highPD)

      # Determine centerpoints of high PD grid cells
      centroids <- sf::st_centroid(cube_mp)

      # Calculate % of high PD grid cell centroids that intersect with
protected areas
      intersecting <- sf::st_intersects(pa, centroids)
      n_intersecting <- sum(lengths(intersecting))
      n_total <- nrow(centroids)
      PD_indicator <- (n_intersecting / n_total) * 100
      indicators[[as.character(p)]] <- PD_indicator

      print(paste("The percentage of high PD grid cells within protected areas
for period", p, "is", PD_indicator, "%"))
    }
  }

 } else {
  # If 'period' column is not present, create a single map and calculate the
indicator for all data
  map <- ggplot() +
    geom_sf(data = world_3035, fill = "antiquewhite") +
    geom_sf(data = PD_cube_geo, mapping = aes(fill = PD)) +
    scale_fill_viridis_c(option = "B") +
    geom_sf(data = pa, fill = NA, color = "darkgreen", linewidth = 0.05) +
    coord_sf(xlim = c(bbox_expanded["xmin"], bbox_expanded["xmax"]),
             ylim = c(bbox_expanded["ymin"], bbox_expanded["ymax"]), expand =
FALSE) +
    xlab("Longitude") + ylab("Latitude") +
    ggtitle(paste("Taxon:", taxon, "\n Phylogenetic Diversity")) +
    theme(panel.grid.major = element_line(color = gray(0.5), linewidth = 0.5),
          panel.background = element_rect(fill = "aliceblue"))

  plots <- map

  # Calculate indicator if cutoff is provided
  if (!is.null(cutoff)) {
    PD_cube_geo$PD_high <- ifelse((PD_cube_geo$PD > cutoff), 1, 0)
    cube_highPD <- PD_cube_geo[PD_cube_geo$PD_high == 1, c("OBJECTID",
"CELLCODE", "PD", "geom", "PD_high")]
```

```
    # Convert to multipolygon object
    cube_mp <- convert_multipolygons(cube_highPD)

    # Determine centerpoints of high PD grid cells
    centroids <- sf::st_centroid(cube_mp)

    # Calculate % of high PD grid cell centroids that intersect with protected
areas
    intersecting <- sf::st_intersects(pa, centroids)
    n_intersecting <- sum(lengths(intersecting))
    n_total <- nrow(centroids)
    PD_indicator <- (n_intersecting / n_total) * 100
    indicators[["Overall"]] <- PD_indicator

    print(paste("The percentage of high PD grid cells that fall within
protected areas is", PD_indicator, "%"))
  }

 }

# Return the list of maps and indicators
if (!is.null(cutoff)) {
  return(list(plots, indicators))
# Return the combined map and indicators for each period or overall
} else {
  return(plots)  # Return only the combined map if no cutoff
}
 }
```

make_shiny_maps()

This function creates an R-shiny app that can showcase multiple generated PD maps (for separate time periods) stored in the PDindicator object generated by `generate_map_and_indicator()`. This significantly aids with visualizing how PD changes through time, and, when working with modelled occurrence cubes, how PD is predicted to change as a result of future changes in species distribution.

Unset

```
#' Visualizing PD maps for time periods in tabs
#'
#' @param plots A list of PD maps produced by the function
#' generate_map_and_indicator(), named by their time-period.
#' @return An r-shiny app with PD maps in tabs
#' @example make_shiny_maps(plots)
```

```
make_shiny_maps <- function(plots){
# Create Shiny app to display the plots in tabs
ui <- fluidPage(
  titlePanel("Phylogenetic Diversity (PD) Maps by Time Period"),

    # Top bar with help text
    fluidRow(
      column(12, align = "center",
             helpText("Browse through the different time periods to see the PD
indicators.")
      )
    ),
    mainPanel(
      # Use do.call to pass the list of tabs as separate arguments
      do.call(tabsetPanel,
              # Dynamically create a tab for each period
              lapply(names(PDindicator[[1]]), function(period) {
                tabPanel(
                  title = paste("Period", period),
                  plotOutput(outputId = paste0("plot_", period),
                             height = "600px", width = "900px")
                )
              })
      )
    )
  )

server <- function(input, output, session) {

  # Render each plot in a separate output
  lapply(names(plots), function(period) {
    output[[paste0("plot_", period)]] <- renderPlot({
      plots[[period]]
    }, res = 150)
  })
}

# Run the Shiny app
shinyApp(ui = ui, server = server)}
```

2.2 Cloud environment

The `pdindicatoR` package can already be installed on any cloud-based computing platform that supports the installation of R and R-packages. Additionally, we are in the process of developing a Jupyter Notebook detailing the installation of pdindicatoR and a worked-out example workflow (see 3.2). We also aim to provide the functionality to connect to Google Earth Engine and generate interactive PD maps, in order to facilitate easier exploration of the data by the user and the possibility to integrate additional environmental layers of choice.

3. Example workflow

This example shows a basic workflow to use the `pdindicatoR` package to generate an area-wide phylogenetic diversity map for species within order Fagales, for the wider area around Hoge Kempen National Park, the first national park of Flanders. In addition, an indicator is calculated as the percentage of high PD grid cells that is currently safeguarded within the boundaries of protected areas. The spatial visualisation can be used to identify potential directions for future expansion of the National Park.

The package can be installed from the GitHub repository using `remotes::install_github()`

```
Unset
# install.packages("remotes")
remotes::install_github("b-cubed-eu/pdindicatoR")
# Load pdindicatoR package
library(pdindicatoR)
```

Loading in example datasets

The example data can be loaded by running the function `retrieve_example_data()`

```
Unset
ex_data <- retrieve_example_data()
tree <- ex_data$tree
cube <- ex_data$cube
grid <- ex_data$grid
pa <- ex_data$pa
```

In this example, we will use the `pdindicatoR` package to generate a phylogenetic diversity map and calculate the PD indicator for the order Fagales, in the region around the national park Hoge Kempen. The example data consists of a phylogenetic tree for the order Fagales, an occurrence data cube for Fagales *sp*. for the study region, the EEA 1km grid [32] and a

shapefile with the boundaries of Natura 2000 sites [33] (cropped to the study region for faster processing).

```
Unset
plot(tree, cex=0.35,y.lim=50)
tree$tip.label # vector with tip labels
```

	year	eeacellcode	specieskey	species	occurrences	distinctobservers
1	2024	1kmE3996N3087	2880539	Quercus rubra	1	1
2	2024	1kmE3997N3088	3054357	Juglans nigra	2	1
3	2024	1kmE3997N3090	3054368	Juglans regia	1	1
4	2024	1kmE3997N3100	2880539	Quercus rubra	1	1
5	2024	1kmE3997N3104	5333294	Castanea sativa	1	1
6	2024	1kmE3997N3104	8313153	Quercus palustris	1	1

First rows of the example species occurrence datacube.

Matching species in phylogenetic tree and datacube

The function `taxonmatch()` can be used to query the Taxonomic Name Resolution Service and match the leaf labels of the tree with OTL's taxonomic backbone, as well as retrieve the corresponding GBIF id's. The resulting table with matches should be carefully evaluated in order to ensure that matching scores are acceptable and that most species have a corresponding gbif_id. Species that can not be reliably matched or that don't have an associated gbif_id, can not contribute to the PD calculation and should be removed.

```
Unset
matched <- taxonmatch(tree)
head(matched) # Carefully evaluate matches before removing non-matches!
```

	search_string	unique_name	approximate_match	score	ott_id	is_synonym	flags	number_matches	gbif_id	orig_tiplabel
1	alfaroa costaricensis	Alfaroa costaricensis	FALSE	1	199041	FALSE		1	7310550	Alfaroa costaricensis
2	alfaroa guanacastensis	Alfaroa guanacastensis	FALSE	1	199043	FALSE		2	*NA*	Alfaroa guanacastensis
3	alfaroa manningii	Alfaroa manningii	FALSE	1	199049	FALSE		1	4205554	Alfaroa manningii
4	alfaroa williamsii	Alfaroa williamsii	FALSE	1	1064109	FALSE		1	7310534	Alfaroa williamsii
5	alfaropsis	Alfaropsis	FALSE	1	200433	FALSE	barren	2	*NA*	Alfaropsis
6	allocasuarina acutivalvis	Allocasuarina acutivalvis	FALSE	1	769753	FALSE		1	2891875	Allocasuarina acutivalvis

Then, the function `append_ott_id()` can be used to append the ott_id's as a new variable to the provided data cube, by joining on gbif_id. When species in the datacube are not included in the provided phylogenetic tree, the ott_id variable will be *NA*. We can use the function

`check_completeness()` to see how complete the provided phylogenetic tree is. Please note that occurrence records for species that are not part of the provided phylogenetic tree will need to be removed. In case this number is large, please consider searching for a more complete phylogenetic tree that covers all your species.

```
Unset
matched_nona <- matched %>% dplyr::filter(!is.na(gbif_id))
mcube <- append_ott_id(tree, cube, matched_nona)
check_completeness(mcube) # Evaluate completeness of tree
mcube <- mcube %>% dplyr::filter(!is.na(ott_id))
```

Calculate Phylogenetic Diversity for each grid cell

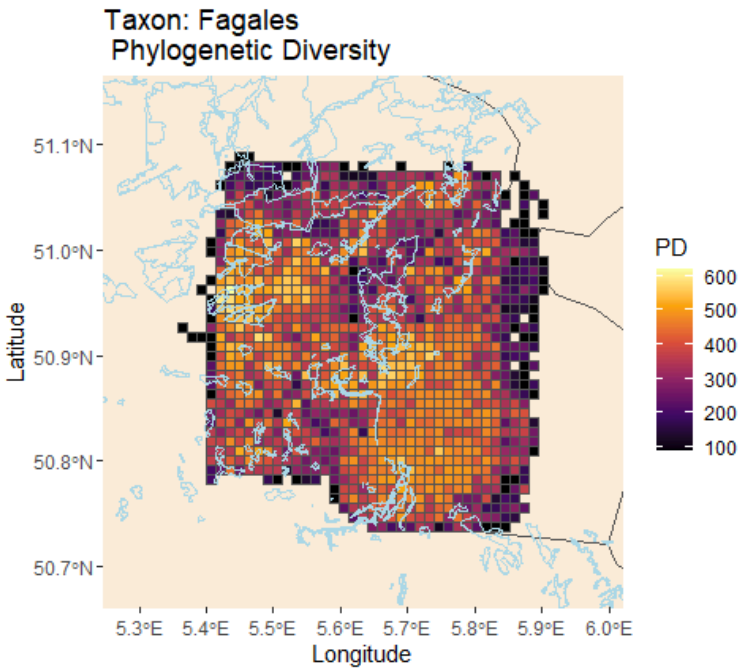
The function `aggregate_cube()` can be used in order to get a list of observed species for each grid cell. The optional argument `timegroup` can be used to indicate a time interval for which the PD metrics should be calculated, eg. `timegroup=5` calculates PD for all occurrences observed within a timespan of five years and produces a seperate map and indicator for each period. If no `timegroup` argument is specified, all occurrences in the dataset will be aggregated over time. Once we have the aggregated cube, the PD metric for all grid cells can be calculated by using the `purrr:map` function to apply the function `get_PD()` for each grid cell. The PD values will be appended to the datacube as a new variable `PD`.

```
Unset
aggr_cube <- aggregate_cube(mcube)
PD_cube <- aggr_cube %>% mutate(PD = unlist(purrr::map(orig_tiplabels, ~
get_pd(tree, unlist(.x)))))
```

Generate PD map & calculate indicator

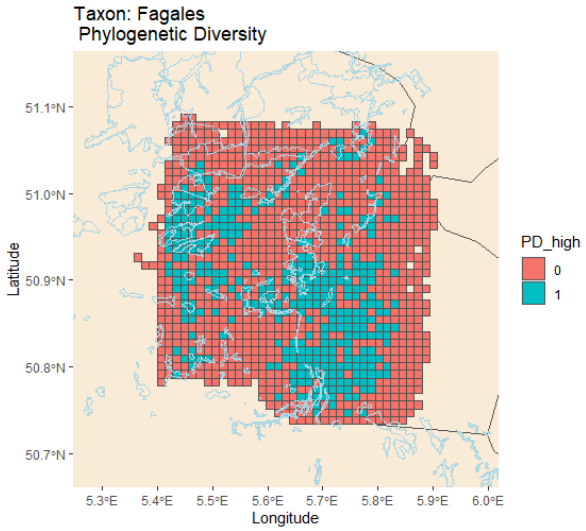
The function `generate_map_and_indicator()` merges the geometries of the grid with the occurrence cube by joining on the `eeaCellCode` field. A map is then produced for the geographic area used to generate the occurrence cube, visualizing phylogenetic diversity using a gradient colour scale. If more zoomed-in maps are desired, the optional argument `bbox_custom` can be used to delineate the bounding box. Coordinates for the desired geographic area can be determined using https://epsg.io/ and selecting the CRS of the used grid. The map is overlaid with a polygon layer depicting the boundaries of Natura 2000 protected areas.

```
Unset
PDindicator <- generate_map_and_indicator(PD_cube, grid, "Fagales")
PDindicator
```



If the `cutoff` argument is specified when calling `generate_map_and_indicator()`, then in addition to the PD map, the percentage of high PD (> cut-off value) cells that fall within protected areas will be calculated. This indicator will be stored together with the continuous PD map and a dichotomized high/low PD map in a list.

```
Unset
PDindicator <- generate_map_and_indicator(PD_cube, grid, "Fagales", cutoff =
450)
indicator <- PDindicator[[2]]
maps <- PDindicator[[1]] # % grid cells within protected areas
print(maps[1]) # PD map
print(maps[2]) # PD low/high map
```




```
[1] "The percentage of high PD grid
cells that fall within protected
areas is 23.46%"
```

4. Discussion

4.1 Challenges

In order to calculate phylogenetic diversity metrics, we need to integrate data on species occurrences with data on the evolutionary relationships between the species of interest.
Phylogenetic trees can be constructed based on genetic analyses or carbon dating studies, and the resulting trees will have branch lengths that either represent gene differences (mutations) or evolutionary time units. Phylogenetic diversity can be calculated based on both these kind of trees, and within conservation biology both methods have been used frequently.
However, since most studies do not analyse specimens of all species within a focus taxon, many published trees are not comprehensive. The Open Tree of Life initiative aimed to combine the information from all the published trees to create one large synthetic Tree of Life for all life on earth. However, since published trees are not in complete accordance in terms of the phylogenetic position of taxa and/or evolutionary distance, the synthetic tree of life does not contain branch lengths [34]. It can therefore currently not be used to calculate phylogenetic diversity metrics that require estimates of evolutionary distance. Finding a published tree with branch lengths that covers all or most of the study species one is interested in, requires some research as well as some trial-and-error. In addition, published tree tip labels often vary significantly in format due to the lack of a naming convention, which makes it difficult to write code that can deal with every possible variation. The taxonomic name resolution service for the Open Tree of Life Taxonomy [35] can only match a limited set of patterns used and often tip labels need to be cleaned up with the appropriate regular expression functions before they can be used as input for the workflow.

It is important to state that the utility and robustness of any phylogenetic diversity indicators currently is fundamentally contingent on the quality and reliability of the underlying phylogenetic tree. For many taxa, there is still no consensus on the exact phylogenetic position along the life tree due to constraints like data availability, uncertainty about taxonomic resolution, and the

methodological choices used to construct phylogenetic trees [34, 36, 37]. This highlights the critical need to address gaps in taxonomic and phylogenetic knowledge. Without such advancements, the applicability of PD indicators in guiding policy and conservation priorities remains constrained, emphasizing the importance of continued investment in phylogenetic research and data integration.

4.2 Future development

The software will be improved based on further testing and feedback from the partners. The main place to collect this feedback is the issue tracker on the pdindicatoR Github repository. In the coming months, the software and indicator will be tested in the case-studies in WP6 and feedback from this process will be implemented in new versions of the software.
We aim to extend the range of available PD metrics the software can calculate (e.g. standard effect size of PD, phylogenetic endemism, PD loss). We will also produce a notebook that can run on a range of cloud infrastructures, which imports `pdindicatoR` and uses its functions in an example workflow. We aim to provide the functionality to connect to Google Earth Engine and generate interactive PD maps, in order to facilitate easier exploration of the data by the user and the possibility to integrate additional environmental layers of choice.

Indicator variability and uncertainty has not been discussed here and will be explored in the future in light of Task 5.4. This is imperative for robust decision-making, highlighting significant trends, and ensuring the reliability and comparability of results across studies. In general, the idea is to perform bootstrap resampling of the species occurrence data cube, but more concrete analyses are necessary. Functions related to uncertainty calculation and interpretation will be provided in a separate R-package that can be used by `pdindicatoR` as dependency.
Occurrence cube indicator software (`R`-packages) will be united within a single framework in the form of a meta-package (Task 5.5). This integration will streamline workflows, enhance usability, and ensure consistency across indicator calculations. It will also facilitate collaboration, improve compatibility between packages, and simplify updates and maintenance, benefiting both developers and end-users. Since we already developed `pdindicatoR` as a functional `R`-package, integration into this meta-package will be straight-forward.

Acknowledgements

We would like to express our gratitude to Dr. Rikki Gumbs (IUCN Phylogenetic Diversity Task Force), Shawn Laffan and Vladimir Mikryukov for providing helpful feedback during the early stages of the design of our indicator. We also thank Christophe van Neste for his continued support with the deployment of `pdindicatoR` on cloud infrastructures.

References

[1] Chao, A., Chiu, CH., Jost, L. Phylogenetic Diversity Measures and Their Decomposition: A Framework Based on Hill Numbers. In: Pellens R, Grandcolas P, editors. Biodiversity Conservation and Phylogenetic Systematics [Internet]. Cham: Springer International Publishing; 2016 [cited 2023 Jun 12]. (Topics in Biodiversity and Conservation; vol. 14). https://doi.org/10.1007/978-3-319-22461-9_8

[2] Faith, DP. Conservation evaluation and phylogenetic diversity. Biological Conservation.1992;61(1):1–10.

[3] Lean C, Maclaurin J. The Value of Phylogenetic Diversity. In: Pellens R, Grandcolas P, editors. Biodiversity Conservation and Phylogenetic Systematics [Internet]. Cham: Springer International Publishing; 2016 [cited 2023 Jun 12]. p. 19–37. (Topics in Biodiversity and Conservation; vol. 14). Available from: http://link.springer.com/10.1007/978-3-319-22461-9_2

[4] Moore JC. Diversity, Taxonomic versus Functional. In: Encyclopedia of Biodiversity [Internet]. Elsevier; 2013 [cited 2023 Jul 3]. p. 648–56. Available from: https://linkinghub.elsevier.com/retrieve/pii/B9780123847195000368

[5] Kiester AR. Species Diversity, Overview. In: Encyclopedia of Biodiversity [Internet]. Elsevier; 2013 [cited 2023 Jul 3]. p. 706–14. Available from: https://linkinghub.elsevier.com/retrieve/pii/B9780123847195001337

[6] Potter KM., Woodall CW. Trends over time in tree and seedling phylogenetic diversity indicate regional differences in forest biodiversity change. Ecological Applications. 2012;22(2):517–31.

[7] Gross N., Bagousse-Pinguet YL., Liancourt P., Berdugo M., Gotelli NJ., Maestre FT. Functional trait diversity maximizes ecosystem multifunctionality. Nat Ecol Evol. 2017 Apr 18;1(5):0132.

[8] Violle C., Navas M.L, Vile D., Kazakou E., Fortunel C., Hummel I., et al. Let the concept of trait be functional! Oikos. 2007 May;116(5):882–92.

[9] Díaz S., Purvis A., Cornelissen JHC., Mace GM., Donoghue MJ., Ewers RM., et al. Functional traits, the phylogeny of function, and ecosystem service vulnerability. Ecol Evol. 2013 Sep;3(9):2958–75.

[10] Flynn DFB., Mirotchnick N., Jain M., Palmer MI., Naeem S. Functional and phylogenetic diversity as predictors of biodiversity- Ecosystem-function relationships. Ecology. 2011;92(8):1573–81.

[11]. Prinzing A., Durka W., Klotz S., Brandl R. The niche of higher plants: evidence for phylogenetic conservatism. Proc R Soc Lond B. 2001 Nov 22;268(1483):2383–9.

[12] Donoghue MJ. A phylogenetic perspective on the distribution of plant diversity. Proc Natl Acad Sci USA. 2008 Aug 12;105(supplement_1):11549–55.

[13] Cavender-Bares J., Kozak KH., Fine PVA., Kembel SW. The merging of community ecology and phylogenetic biology. Ecology Letters. 2009 Jul;12(7):693–715.

[14] Cadotte MW., Cardinale BJ., Oakley TH. Evolutionary history and the effect of biodiversity on plant productivity. Proc Natl Acad Sci USA. 2008 Nov 4;105(44):17012–7.

[15] Cadotte MW., Dinnage R., Tilman D. Phylogenetic diversity promotes ecosystem stability. Ecology. 2012 Aug;93(sp8):S223–33.

[16] Pu Z., Daya P., Tan J., Jiang L. Phylogenetic diversity stabilizes community biomass. Journal of Plant Ecology. 2014 Apr 1;7(2):176–87.

[17] Cadotte, M.W. Phylogenetic diversity and productivity: gauging interpretations from experiments that do not manipulate phylogenetic diversity. Funct Ecol. 2015; 29: 1603-1606. https://doi.org/10.1111/1365-2435.12543

[18]. Venail P, Gross K.., Oakley TH., Narwani A., Allan E., Flombaum P., et al. Data from: Species richness, but not phylogenetic diversity, influences community biomass production and temporal stability in a re-examination of 16 grassland biodiversity studies [Internet]. Dryad; 2016 [cited 2023 Jul 14]. p. 174979 bytes. Available from: https://datadryad.org/stash/dataset/doi:10.5061/dryad.s2h01

[19] Pereira HM, Ferrier S.., Walters M., Geller GN., Jongman RHG., Scholes RJ., et al. Essential Biodiversity Variables. Science. 2013 Jan 18;339(6117):277–8.

[20] Tucker CM., Cadotte MW., Carvalho SB., Davies TJ., Ferrier S., Fritz SA., et al. A guide to phylogenetic metrics for conservation, community ecology and macroecology: A guide to phylogenetic metrics for ecology. Biol Rev. 2017 May;92(2):698–715.

[21] Faith, D. Valuation and Appreciation of Biodiversity: The “Maintenance of Options” Provided by the Variety of Life. Frontiers in Ecology and Evolution. 2021; 9. https://doi.org/10.3389/fevo.2021.635670.

[22] IPBES. The IPBES regional assessment report on biodiversity and ecosystem services for Asia and the Pacific. Karki, M., Senaratna Sellamuttu, S., Okayasu, S., and Suzuki, W. (eds). Secretariat of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services, Bonn, Germany; 2018. 612 pages.

[23] Robuchon, M., da Silva, J., Dubois, G., Gumbs, R., Hoban, S., Laikre, L., Owen, N. R., & Perino, A. Conserving species' evolutionary potential and history: Opportunities under the Kunming–Montreal Global Biodiversity Framework. Conservation Science and Practice. 2023; 5(6). https://doi.org/10.1111/csp2.12929

[24] Mikryukov V., Abarenkov K., Laffan S., Robertson T., McTavish EJ., Jeppesen TS., Waller J., Blissett M., Kõljalg U., Miller JT. PhyloNext: A pipeline for phylogenetic diversity analysis of GBIF-mediated data. BMC Ecology and Evolution. 2024; 24(1), 76. https://doi.org/10.1186/s12862-024-02256-9

[25] Redelings BD., Holder MT. "A supertree pipeline for summarizing phylogenetic and taxonomic information for millions of species." PeerJ. 2017;5:e3058. https://doi.org/10.7717/peerj.3058

[26] Laffan, S. W., Lubarsky, E. and Rosauer, D. F. Biodiverse, a tool for the spatial analysis of biological and related diversity. – Ecography. 2010; 33: 643–647 [Version 0.13].

[27] P. Di Tommaso, et al. Nextflow enables reproducible computational workflows. Nature Biotechnology. 2017; 35, 316–319. https://doi.org/10.1038/nbt.3820

[28] Merkel D. Docker: lightweight linux containers for consistent development and deployment. Linux journal. 2014; 2014(239):2.

[29] Laffan, S. W., Tomczyk, J. BiodiverseR: An R interface to the Biodiverse engine for the spatial analysis of diversity [Source code, version: 0.0.0.9001]. https://github.com/shawnlaffan/BiodiverseR

[30] Breugelmans, L., Trekels, M., & Hendrickx, L. pdindicatoR: Calculate and visualize a phylogenetic diversity indicators based on species occurence data cubes (Version 0.0.2) [Computer software]. https://github.com/b-cubed-eu/pdindicatoR/

[31] Faith, D. P., & Baker, A. M. Phylogenetic diversity (PD) and biodiversity conservation: some bioinformatics challenges. Evolutionary bioinformatics. 2006; 2. https://doi.org/10.1177/117693430600200007

[32] European Environment Agency. EEA reference grid for Belgium (1km) [Dataset]. European Environment Agency. 2024. https://www.eea.europa.eu

[33] UNEP-WCMC and IUCN. Protected Planet: The World Database on Protected Areas (WDPA) [Dataset]. 2024. www.protectedplanet.net

[34] Hinchliff, C.E., Smith, S.A., Allman, J.F., Burleigh, J.G. , Chaudhary, R., Coghill, L.M., Crandall, K.A., Deng, J. , Drew, B.T., Gazis, R., Gude, K., Hibbett, D.S., Katz, L.A. , Laughinghouse, H.D. , McTavish, E.J., Midford, P.E., Owen, C.L., Ree, R.H., Rees, J.A., Soltis, D.E. , Williams, T., Cranston, K.A. Synthesis of phylogeny and taxonomy into a comprehensive tree of life, Proc. Natl. Acad. Sci. U.S.A. 2015; 112 (41) 12764-12769, https://doi.org/10.1073/pnas.1423041112

[36] Smith, S. A., and J. W. Brown. Constructing a broadly inclusive seed plant phylogeny. American Journal of Botany. 2018; 105(3): 302–314.

[37] Sanderson, MJ. Estimating absolute rates of molecular evolution and divergence times: a penalized likelihood approach. Mol Biol Evol. 2002 Jan;19(1):101-9. https://doi.org/10.1093/oxfordjournals.molbev.a003974