Paleobiodiversity Analysis

```
library(paleoDiv)
#> Loading required package: ape
#> Loading required package: stringr
```

Basic Workflow of the paleoDiv Package

We can start the analysis by downloading some occurrence data from the Paleobiology Database, in this case for Stegosaurs:

```
pdb("Stegosauria")->Stegosauria
```

The resulting occurrence dataset should look somewhat like this (but with more columns and many more rows):

```
knitr::kable(head(Stegosauria[,c(c(1:7),18,20:21)]))
```

occurrence_no	record_type	reid_no	flags	collection_no	identified_name	identified_rank	tna	lag	eag
149599	occ	NA		13214	Stegosaurus sp.	genus	Stegosaurus sp.	145.0	157.3
219973	occ	12664		22648	Stegosaurus stenops	species	Stegosaurus stenops	145.0	157.3
220057	occ	NA		22678	Stegosaurus sp.	genus	Stegosaurus sp.	145.0	155.7
220164	occ	NA		22711	Stegosaurus sp.	genus	Stegosaurus sp.	145.0	152.1
225585	occ	NA		21852	Stegosaurus sp.	genus	Stegosaurus sp.	145.0	150.8
260839	occ	23241		25221	Stegosauria ? indet.	unranked clade	Stegosauria ? indet.	93.5	105.3

The columns that are of primary interest here are occurrence_no, tna (which at this point is simply a copy of identified_name), eag and lag (which contain the maximum and minimum ages of the occurrence).

Since the identified_name/tna column at this point will contain a fair deal of common character combinations leading to an overinflation of unique values (e.g. cf., gen. nov., sp. nov., extra spaces etc), it is recommended to run the following:

```
occ.cleanup(Stegosauria)->Stegosauria$tna
#> [1] "94 factor levels reduced down to 71"
```

This replaces the tna-column with a filtered version, with such common character strings removed. The function also prints a message giving information about the reduction in unique factor levels of tna. In this case 94 are reduced to 71. This reduces the number of duplicate taxa in the dataset, and thus will help provide more accurate absolute diversity estimates down the line. However, if complete taxonomic accuracy is the goal, then manually checking the dataset (e.g. after the next step, see below) may become necessary.

If we are interested in analyzing the pattern of paleobiodiversity, our next step in the paleoDiv workflow is to build a taxon-range table, which is simply a data.frame() containing the minimum and maximum ages for each unique factor value in tna (or, if manually modified, any other column or vector). We do this using the mk.sptab() function.

```
mk.sptab(Stegosauria)->sptab_Stegosauria
```

By default, mk.sptab() takes an occurrence dataset as input and creates a range table containing one row for each unique factor level in the column tna. However, these settings are freely modifiable, making the function applicable to any columns in a data.frame() or even individual vectors containing the maximum and minimum ages and the taxon names (or other category).

Our generated taxon-range table should now look like this:

tna	max	min	ma
Adratiklit boulahfa	167.7	163.5	165.6
Alcovasaurus longispinus	157.3	152.1	154.7
Amargastegos brevicollus	129.4	125.0	127.2
Bashanosaurus primitivus	168.3	163.5	165.9
Changdusaurus laminaplacodus	174.1	163.5	168.8
Chialingosaurus kuani	168.3	163.5	165.9

For convenience, the function pdb.autodiv() combines all of these steps into a single function, and can also be used with a vector of several taxon names as input:

This represents the most convenient method of quickly downloading occurrence data and constructing taxon-range tables for multiple clades in the paleoDiv package. The output of pdb.autodiv is a list() object containing multiple data.frames(): one for each occurrence dataset downloaded from the Paleobiology database, and one for each taxon's taxon-range table, with the prefix "sptab_".

Having generated a taxon-range table, we can now use it to estimate and plot diversity using the function divdistr_():

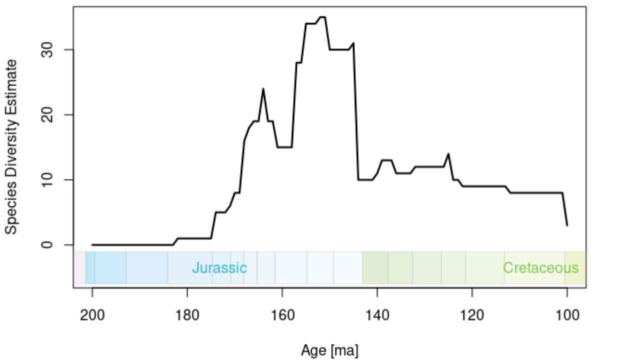
```
divdistr_(150,table=sptab_Stegosauria)
#> [1] 30
```

This tells us that at there are 30 species (or other lowest identified taxonomic levels) in our taxon-range table whose stratigraphic ranges include 150 ma. This function can be applied to an entire vector of geological ages, if we are interested in how diversity changed over time:

```
divdistr_(c(170:120),table=sptab_Stegosauria)
#>  [1]  8  8 16 18 19 19 24 19 19 15 15 15 15 28 28 34 34 34 35 35 30 30 30 30 30
#> [26] 31 10 10 10 10 11 13 13 13 11 11 11 11 12 12 12 12 12 12 12 14 10 10  9  9
#> [51]  9
```

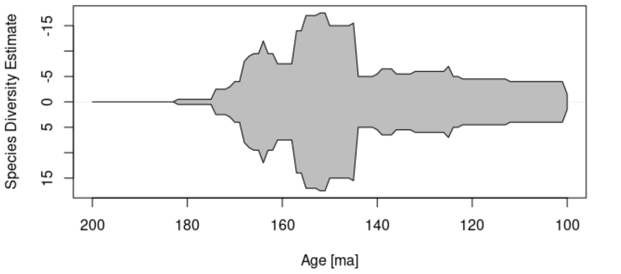
And it can also be used as a function for plotting these data, e.g. using curves():

```
curve(divdistr_(x,sptab_Stegosauria), xlim=c(200,100),ylim=c(-5,35), lwd=2,xlab="Age
        [ma]",ylab="Species Diversity Estimate")
#to add a geological timescale, we can use ts.stages() and ts.periods():
ts.stages(ylim=c(-6,-1),alpha=0.3,border=add.alpha("grey",0.3))
ts.periods(ylim=c(-6,-1),alpha=0.0)
```



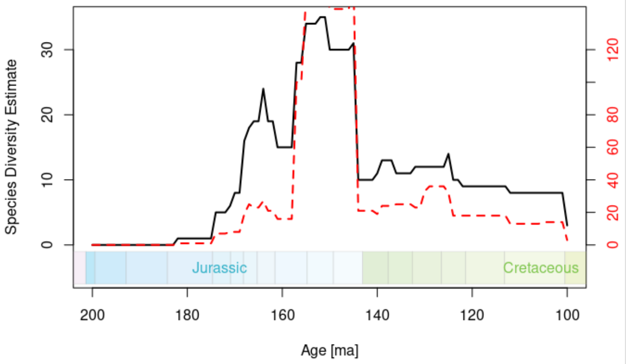
…or as a spindle diagram, using the viol()-function provided with this package, by providing divdistr_ for its stat parameter, overriding its default binding to the density() function:

```
viol(c(100:200), pos=0, stat=divdistr_, table=sptab_Stegosauria,
        lim=c(200,100),add=F,ylab="Species Diversity Estimate",xlab="Age [ma]")
```



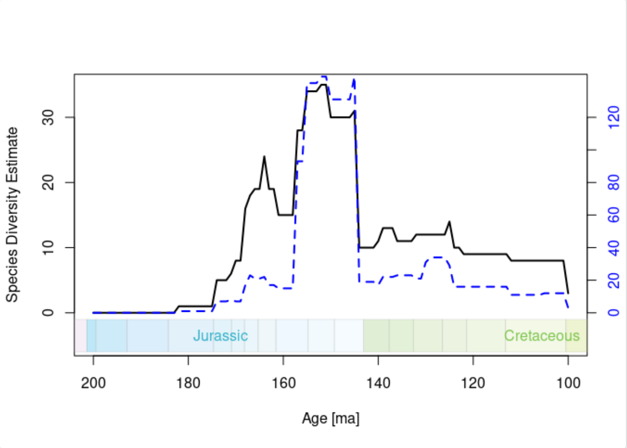
We can also plot abundance on the same graph using the abdistr_() function (which works much like divdistr_(), but by default uses occurrence datasets instead of taxa), e.g. as follows:

```
curve(divdistr_(x,sptab_Stegosauria), xlim=c(200,100),ylim=c(-5,35), lwd=2,xlab="Age
        [ma]",ylab="Species Diversity Estimate")
#to add a geological timescale, we can use ts.stages() and ts.periods():
ts.stages(ylim=c(-6,-1),alpha=0.3,border=add.alpha("grey",0.3))
ts.periods(ylim=c(-6,-1),alpha=0.0)

curve(abdistr_(x,Stegosauria)/4, xlim=c(200,100),ylim=c(-5,35),col="red", lwd=2,lty=2,add=T)
axis(4,at=seq(0,30,5), lab=seq(0,30,5)*4, col.axis="red")
```



If we are interested in how collection intensity/the number of collections compares to taxonomic diversity or abundance, we can use pdb() to download collections data instead of occurrence data:

```
pdb("Stegosauria", what="colls")->Stegosauria_colls
curve(divdistr_(x,sptab_Stegosauria), xlim=c(200,100),ylim=c(-5,35), lwd=2,xlab="Age
        [ma]",ylab="Species Diversity Estimate")
#to add a geological timescale, we can use ts.stages() and ts.periods():
ts.stages(ylim=c(-6,-1),alpha=0.3,border=add.alpha("grey",0.3))
ts.periods(ylim=c(-6,-1),alpha=0.0)

#now plot the number of collections alongside the diversity curve
curve(abdistr_(x,Stegosauria_colls)/4, xlim=c(200,100),ylim=c(-5,35),col="blue",
        lwd=2,lty=2,add=T)
axis(4,at=seq(0,30,5), lab=seq(0,30,5)*4, col.axis="blue")
```

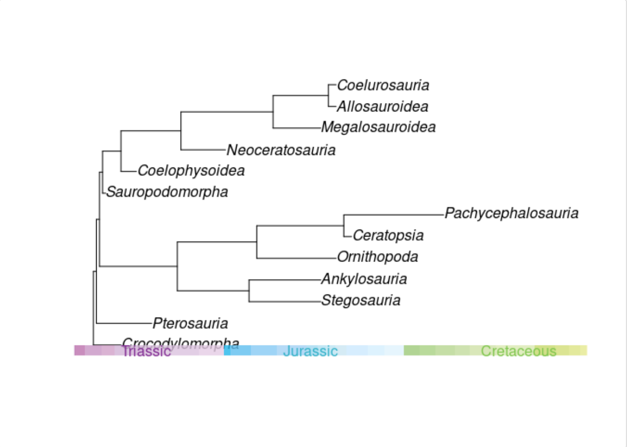


For the next examples, a calibrated phylogeny of Archosauria, a matrix used for its calibration and a list() object containing multiple taxon-range tables are provided with the package as example data:

```
data(archosauria)
data(tree_archosauria)
data(ages_archosauria)
```

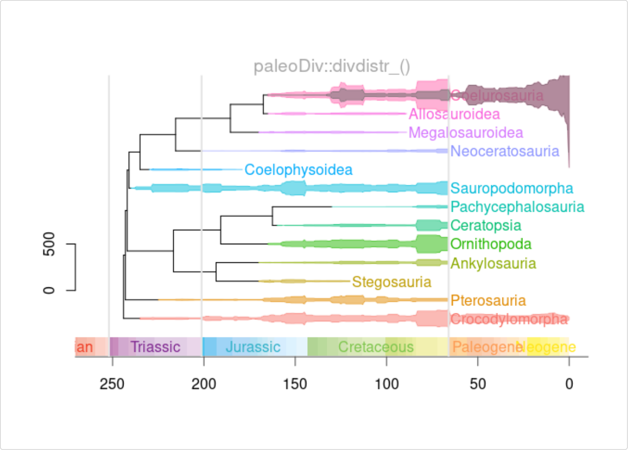
We can plot the phylogeny using ape:

```
ape::plot.phylo(tree_archosauria)
ts.stages(tree_archosauria,alpha=0.8)
ts.periods(tree_archosauria, names=T, alpha=0)
```



One of the key functions of paleoDiv is phylo.spindles(), which is optimized for plotting diversity as a spindle-diagram relative to phylogeny:

```
phylo.spindles(phylo0=tree_archosauria,occ=archosauria,col=add.alpha(ggcol(13)),ages=ages_archosauria,txt.y=.5,
        dscale=0.005,xlim=c(260,0),axis=F,tbmar=c(1.5,.5),txt.x=c(66,ages_archosauria[2:12,"LAD"],66))
#add a timescale
ts.stages(tree_archosauria, ylim=c(-1,0),alpha=0.8)
ts.periods(tree_archosauria, names=T, ylim=c(-1,0),alpha=0)

#add an x axis with custom tick positions:
axis(1,at=tsconv(seq(300,-50,-50),tree_archosauria), lab=seq(300,-50,-50),
        cex=0.75,col="grey30",col.lab="grey30")
#add a short y axis serving as a scale bar
axis(2, at=c(500,1000)*0.005, lab=c(0,500))

#add vertical lines marking major mass extinctions.
abline(v=tsconv(c(252,201.3,66),tree_archosauria),lwd=2, col="grey90")

mtext(side=3, cex=1.2,"paleoDiv::divdistr_()",col="darkgrey")

#we can manually add spindles anywhere using viol(), e.g. to plot the diversity of important
        subtaxa, in this case that of birds:
viol(x=c(260:0),stat=divdistr_, pos=13,
        table=convert.sptab(archosauria$sptab_Aves,tree_archosauria),dscale=0.005,
        cutoff=tsconv(c(165,0),tree_archosauria),fill=add.alpha("grey40"),
        col=add.alpha("grey40"))
```

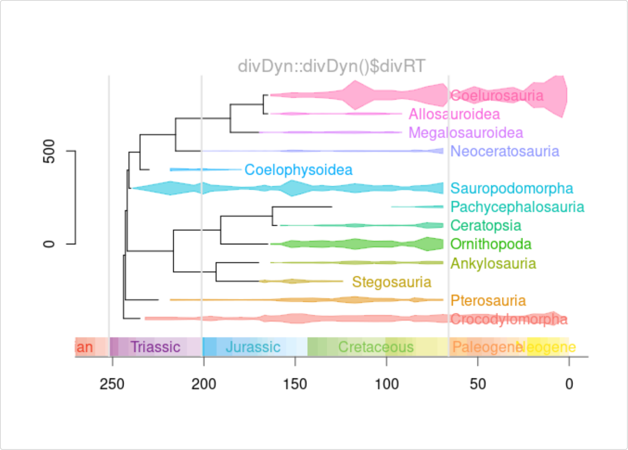


Important parameters for phylo.spindles include: * phylo0 The phylogeny to be plotted. This should be time-calibrated (e.g. using strap:datePhylo()) * occ The dataset to be used for plotting. This has to either be a list() containing data.frames to be used as an argument for the function divdistr_() (or any other function it is overridden with) as parameter “table”, or a data.frame() or matrix() containing an x column and columns matching the names of the phylogenetic tree’s tip.labels. * ages Optional data matrix used for cropping each spindle to the known stratigraphic range of the taxon. If provided, this should take the form of a matrix with row names being the same as the trees tip.labels, and two columns named “FAD” and “LAD” giving minimum and maximum geological ages for each taxon. * dscale Vertical scaling parameter for the spindles. May need manual adjustment, depending on the desired scale. * txt.x Either a single number of a vector of the same length as the number of tree tips giving the horizontal position for labels (if labels==TRUE). In this case, the second column of ages is used, placing the labels at the end of the spindles, but the first and last values are replaced by 66 in order to place labels within the plot boundaries * For further details, see ?phylo.spindles

Note that ape::plot.phylo(), which is used by this function for generating the tree, plots trees to a time coordinate counting forward from the root age. As a result, we need to convert all geological ages to this coordinate system by subtracting them from the root age of the phylogeny (facilitated by the tsconv()-function which takes ages and a tree as input).

By replacing occ with a data.frame() or matrix(), we can also use phylo.spindles() to plot a variety of other time series data, such as morphological disparity, abundance, number of occurrences, diversity estimates made using different methods and/or packages, etc. As an example, the object diversity_table contains a time series of by-stage diversity estimates for the same taxa made using the divDyn packages (in this case divRT):

```
data(diversity_table)

phylo.spindles(tree_archosauria,occ=diversity_table,ages=ages_archosauria,txt.y=.5,xlim=c(260,0),axis=F,dscale=0.01,col=add.alpha(ggcol(13)),tbmar=c(1.5

#add a timescale
ts.stages(tree_archosauria, ylim=c(-1,0),alpha=0.8)
ts.periods(tree_archosauria, names=T, ylim=c(-1,0),alpha=0)

#add an x axis with custom tick positions:
axis(1,at=tsconv(seq(300,-50,-50),tree_archosauria), lab=seq(300,-50,-50),
        cex=0.75,col="grey30",col.lab="grey30")
#add a short y axis serving as a scale bar
axis(2, at=c(500,1000)*0.01, lab=c(0,500))

#add vertical lines marking major mass extinctions.
abline(v=tsconv(c(252,201.3,66),tree_archosauria),lwd=2, col="grey90")

mtext(side=3, cex=1.2,"divDyn::divDyn()$divRT",col="darkgrey")
```


divDyn::divDyn()$divRT
Coelurosauria
Allosauroidea
Megalosauroidea
Neoceratosauria
Coelophysoidea
Sauropodomorpha
Pachycephalosauria
Ceratopsia
Ornithopoda
Ankylosauria
Stegosauria
Pterosauria
Crocodylomorpha
an
Triassic
Jurassic
Cretaceous
Paleogene
Neogene