---

title: \Huge \textbf{Species sympatry constrains brain size evolution in Primates}

# date: "`r format(Sys.time(), '%B %d, %Y')`"

#I follow: https://stackoverflow.com/questions/52918716/authors-and-affiliations-in-the-yaml-of-rmarkdown for author display

author:

- name: \textit{Benjamin Robira}

email: benjamin.robira@normalesup.org

institute: [CEFE, MH]

correspondence: true

- name: \textit{Benoit Perez-Lamarque}

email: benoit.perez@ens.psl.eu

institute: [ENS, MNHN]

correspondence: true

institute:

- CEFE: Centre d’Écologie Fonctionnelle et Évolutive, Université de Montpellier & CNRS, Montpellier, France.

- MH: Anthropologie et Ethnobiologie, Centre National de la Recherche Scientifique/Muséum National d'Histoire Naturelle, University Paris Diderot, Sorbonne Paris Cité, Musée de l'Homme, Paris, France

- ENS: Institut de Biologie de l’École Normale Supérieure (IBENS), École Normale Supérieure, CNRS, INSERM, Université PSL, Paris, France

- MNHN: Institut de Systématique, Évolution, Biodiversité (ISYEB), Muséum National d’Histoire Naturelle, CNRS, Sorbonne Université, EPHE, Université des Antilles, Paris, France

output:

# - '--lua-filter=scholarly-metadata.lua'

# - '--lua-filter=author-info-blocks.lua'

bookdown::pdf\_book:

#bookdown::word\_document2:

number\_sections: false

toc: false

#citation\_package: natbib

latex\_engine: pdflatex

fig\_caption: true

pandoc\_args:

- '--lua-filter=scholarly-metadata.lua'

- '--lua-filter=author-info-blocks.lua'

bibliography: bibliographyarticlepackage.bib

csl: science.csl

#biblio-style: apa

always\_allow\_html: true

link-citations: yes

urlcolor: blue

fontsize: 12pt

header-includes:

# - \usepackage[nolists, nomarkers,tablesfirst]{endfloat} # For figures and tables at end

- \usepackage{lineno} # For line numbering

- \linenumbers # For line numbering

- \usepackage{setspace}\doublespacing

- \setlength{\parskip}{0em} #to remove line gaps

- \DeclareUnicodeCharacter{2212}{-}

- \usepackage{caption}

- \captionsetup[figure]{font=small}

- \newcommand{\beginsupplement}{

\setcounter{table}{0}

\renewcommand{\thetable}{S\arabic{table}}

\setcounter{figure}{0}

\renewcommand{\thefigure}{S\arabic{figure}}

}

---

```{r setup, include=FALSE, echo=FALSE, message=FALSE}

knitr::opts\_chunk$set(include=FALSE, echo=FALSE, message=FALSE)

knitr::opts\_chunk$set(dpi=300) # Figure resolution and size out.width = '100%',

knitr::opts\_chunk$set(fig.pos = 'p'#, fig.align = 'center'

) # Places figures on pages separate from text, centered

knitr::opts\_chunk$set(fig.env="figure") # Latex figure environment

knitr::opts\_knit$set(eval.after = "fig.cap") #To insert R code into R figure caption

```

```{r, echo=FALSE, results= 'hide'}

#Import librairies

library(readr)

#Plot

library(RColorBrewer)

library(tidyr)

library(stringr)

library(svMisc)

library(plotrix)

library(circlize)

#Spatial

library(rworldmap) # World map

library(cleangeo) #to clean it otherwise issues with intersection

library(maps)

library(rgeos) #for readOGR; gArea/gCentroid...

library(sf) #for intersection

library(rgdal)

library(geosphere)

#Phylogeny

library(phytools)

library(ape)

library(phylolm)

#Segmentation

library(strucchange)

#Import own function

source("T:/Saved\_PhD/Empirical\_analysis/Scripts&Functions/Functions/toolbox.R", local = knitr::knit\_global())

#Create citation fusion between articles and package (based on toolbox function)

citeR(

bibliographyArticle="C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Article/bibliographyarticle.bib",

bibliographyOutput="C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Article/bibliographyarticlepackage.bib",

rgeos,

geosphere,

phytools,

geiger,

RPANDA,

caper,

neurobase,

misc3d,

phylolm,

nlme,

MCMCglmm,

coda,

strucchange

)

#Load environments

load("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/REnvironments/Data\_spatial\_primate.RData")

load("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/REnvironments/geography\_traits\_biogeobears.RData")

load("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/REnvironments/PGLSdiversification\_withautocorr.RData")

load("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/REnvironments/PGLSdirectionSelection.RData")

```

\captionsetup{list=no}

<!-- The front page -->

<!-- \centering -->

<!-- \raggedright -->

<!-- \newpage -->

<!-- \tableofcontents -->

\newpage

\*\*Abstract:\*\* The diversity in animal cognition raises the question of its underlying evolutionary drivers. Selection upon more advanced cognitive abilities can stem from interactions of individuals with conspecifics within the social unit, among generations, between social units, or with the rest of their environment. Indeed, one species rarely occupies an area alone: Space is often shared between many species from the same ecological guild that can interact directly or indirectly. For instance, primate species occupying a similar dietary niche might compete for food access, which can induce a cognitive evolutionary arms race, so as to outperform competitors. Alternatively, the more foraging species, the more uncertainty in resource location and availability, such that at one point, resources might be too unpredictable and larger cognitive abilities might no longer be advantageous. As such, to test whether species co-occurrence shaped current patterns of cognition, we retraced the evolutionary history of multiple primate brain areas involved in foraging activities or not while considering competitive or non-competitive evolutionary scenarios. We found that the evolution of the relative size of areas involved in foraging-related information processing and/or retention, as well as in areas related to processing social information, are better described by models accounting for species co-occurrence within dietary guilds. More precisely for these brain areas, species co-occurrence was associated to a decrease, and never to an increase, of their relative sizes. Coherently with the observed wide variability in sympatry rate and intensity, the degree of encephalisation was unrelated to the evolutionary success of a lineage (i.e. its diversification rate). Overall, this comparative study suggests that species co-occurrence stands as a brake to positive selection towards larger cognitive abilities, yet leaves open the question of the underlying ecological mechanisms at play.

<!-- subject to an arms race among species (\*Red Queen Intelligence Hypothesis\*). -->

\*\*Keywords:\*\* "Brain size - Cognition - Cooperation - Competition - Intelligence evolution - Primates - Species co-occurrence – Diversification "

<!-- Reviewer possibles: Decasien, Powell, Barton, Kamilar, Harmon, Drury -->

```{r wordCount}

words <- RmdWords("Article.Rmd")

```

<!-- TC:ignore -->

\*\*Word Count:\*\* `r words$num\_words` \newline

<!-- Character Count: `r words$num\_char` -->

<!-- TC:endignore -->

\newpage

# Introduction

On the road to brain size evolution, generally considered as an equivalent of cognition evolution, mysteries are plenty [@dunbar2017there]. It remains puzzling why humans have a brain that is that large, relatively to body size, compared to other animals, or why primate brain architecture, in general, is much more complex than observed in other taxa. Overall, the evolution of the brain is the consequence of constraints [e.g. energetic limitations, @navarrete2011energetics], but also of socio-ecological drivers promoting cognitive abilities [@gonzalez2018inference].

| To grasp the rationale of current hypotheses that aim to describe the evolution of cognition [@van2006some; @dunbar2017there], consider a primate individual foraging for food. Primates are pivotal species to study the evolution of cognition given the complexity of their socio-ecological environment and the inevitable implication for retracing human evolutionary history [@byrne2000evolution]. The chosen focal is not looking for any type of food but specific one, as for instance, most primate species often look for fruits. Finding fruits for a primate is not an easy task. In fact, fruits are the archetype of a hard-to-find resource likely to promote cognitive development because fruit trees are rare, dispersed, and do not produce constantly along the year nor between years, albeit their fruiting period remains nonetheless predictable [@janmaat2016spatio]. Additionally, they are energy-rich, allowing expansion of costly tissues such as brain tissues [@clutton1980primates]. Overall, primates thus show remarkable abilities to navigate precisely and target fruit trees likely to yield ripe resource despite they are out of their sensory range [@trapanese2019review], in part because of advanced spatio-temporal information retention [e.g. spatial knowledge: @normand2009forest; @robiraroute, temporal knowledge: @janmaat2006evidence; @janmaat2013chimpanzees; @janmaat2013tai].

| When the resource is out of sight, moving at random could mean travelling unnecessary long distances to finally reach a potentially void tree. Hence, for the focal, being capable of processing immediate environmental cues and clues to know where specifically to find fruit trees [@dall2005information; @grove2013evolution] and whether it is probable that these trees currently yield fruits [@dall2005information; @robirainreview; @janmaat2016spatio] would be a considerable advantage. Having an appropriate cognitive machinery may avoid costly detours and time recursions to food patches adequately. This simple picture draws the basis of the \*Ecological Intelligence Hypothesis\* [@clutton1980primates; @milton1981distribution; @rosati2017foraging] which stipulates that cognition was selected as a way to buffer the spatio-temporal complexity of the environment (e.g. @van2012large).

| In its quest for food, however, the focal might not be alone: For instance, primates often form social groups [@kappeler2002evolution] in which group members spend considerable time together, have established bounds and constantly share information. Thus, they might altogether look for, and process, cues and clues leading to food sources. To do so, being able to process signals emitted by other maneuvers would be an advantage, as well as “reading through their mind” [@devaine2017reading] in order to plan ahead for collective actions or even outsmart them [@byrne1994machiavellian]. This therefore means plenty of additional information to process, and constitutes the socle of the \*Social Intelligence Hypothesis\* [@dunbar2017there].

| In the focal group, however, all individuals are not equally knowledgeable: Perhaps because some have more experience than others, such as the elderly ones would be more experienced than the youths, with the latter thus learning from the former [@laland2004social]. For instance, young individuals may progressively learn how to process a specific tool to access a hidden comestible part as for nuts [@boesch1994nut] by observing and reproducing, or possibly being taught (learning ways detailed in @estienne2019acquisition). In other words, there is vertical transmission that passes by over generations and knowledge thereby accumulates: This is the \*Cultural Intelligence Hypothesis\* [@whiten2007evolution; @reader2002social], also known as the \*Vygotskian Intelligence Hypothesis\* in humans [@herrmann2007humans; @tomasello2009cultural; @wilson1991molecular; @van2011social].

| Finally, while the group peacefully forages, it might sense the past or current presence of another group and anticipate its move, such as wild baboons do when prioritizing food likely to be depleted by other troops [@noser2010wild]. The neighboring groups might indeed be competitors for food, mates or any essential resource, and as much as it is advantageous to be able to read through the environment and intra-group companions, it might be advantageous to be able to decode information relative to other groups’ presence and attribute (e.g. group size), a view brought by the \*Napoleonic Intelligence Hypothesis\* [@ashton2020interactions].

| Yet, in this overly simplistic picture, we moved from a unique individual to a group of individuals and then to multiple groups. Never was it question of multiple species. Yet, space is a place shared between a plethora of species, some of them occupying a same ecological niche. As much as conspecifics could be competitors, or direct or indirect cooperators, so could be individuals from another species with similar ecological preferences, for instance with regards to diet. As such, co-occurrence of species from the same guild might contribute into shaping animal cognition. On the one side, the presence of sympatric species with overlapping diet could contribute in reinforcing selective pressures for advanced cognition because (i, competition) species would compete for food access (i.e. Red Queen paradigm, [@van1973new]). Co-occurrence would increase the environmental complexity due to impoverishment of food and addition of noise to the spatio-temporal availability signal because of unforeseen depletion. (ii, cooperation/exploitation) sympatric species presence cues could also represent additional information to process to infer resource location and availability. Logically then, species with higher brain size would be the most evolutionary successful, thus “booming” and intensively diversifying as for hominins [@melchionna2020macroevolutionary]. On the other side, (i) the increase in environmental complexity could be such that advanced cognition is no longer adaptive (see for instance [@grove2013evolution] and [@robirainreview] for the limit of the adaptiveness of spatial and temporal cognition respectively), or (ii) the additional cues provided by other species presence would not add, but replace, and perhaps be more easily interpreted than, environmental cues of food availability. In this latter case, selective pressure on cognitive abilities would be relaxed, inducing a decrease in brain size in sympatric species compared to lonely species. Following this rationale, species living in co-occurrence or not with other species would not face the same evolutionary fate despite initially similar environmental conditions. Thus, under this scenario, we do not expect the evolutionary success of specific lineages to be related with brain size. In this study, we therefore aimed to test how species co-occurrence contributed to shaping the evolutionary history of the encephalisation of the whole, or part of, the brain, and whether this induced a "boom" or a "brake" to cognitive abilities and associated evolutionary success by focusing on frugivorous primates as a study example.

\hfill

```{r sampleCalculation, warning = FALSE, message = FALSE}

###Set working directory

setwd("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates")

repetition=2\*2\*2\*10#length(frugivoryThresholdVector)\*length(folivoryThresholdVector)\*length(geographicThresholdVector)\*randomSampling

checkSampleFruit <- rep(NA, times=repetition)

checkSampleLeaf <- rep(NA, times=repetition)

checkSampleRange <- rep(NA, times=repetition)

checkSampleBrain <- rep(NA, times=repetition)

checkSampleEQ <- rep(NA, times=repetition)

checkSampleNeocortex <- rep(NA, times=repetition)

checkSampleHippocampus <- rep(NA, times=repetition)

checkSampleCerebellum <- rep(NA, times=repetition)

checkSampleStriatum <- rep(NA, times=repetition)

checkSampleMOB <- rep(NA, times=repetition)

checkSampleRange <- rep(NA, times=repetition)

counter=0

for(a in 1:2){

for(b in 1:2){

for(c in 1:2){

for(d in 1:10){

counter=counter+1

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleFruit",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleFruit[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleLeaf",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleLeaf[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleRange",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleRange[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleBrain",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleBrain[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleEQ",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleEQ[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleNeocortex",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleNeocortex[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleHippocampus",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleHippocampus[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleCerebellum",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleCerebellum[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleStriatum",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleStriatum[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/Sample\_size/checkSampleMOB",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

checkSampleMOB[counter] <- toAdd[1]

}, error=function(e){

#Do nothing

}

)

}

}

}

}

checkSampleFruit <- unlist(checkSampleFruit)

checkSampleLeaf <- unlist(checkSampleLeaf)

checkSampleRange <- unlist(checkSampleRange)

checkSampleBrain <- unlist(checkSampleBrain)

checkSampleEQ <- unlist(checkSampleEQ)

checkSampleNeocortex <- unlist(checkSampleNeocortex)

checkSampleHippocampus <- unlist(checkSampleHippocampus)

checkSampleCerebellum <- unlist(checkSampleCerebellum)

checkSampleStriatum <- unlist(checkSampleStriatum)

checkSampleMOB <- unlist(checkSampleMOB)

#Min values

minFruit <- min(checkSampleFruit)

minLeaf <- min(checkSampleLeaf)

minRange <- min(checkSampleRange)

minBrain <- min(checkSampleBrain)

minEQ <- min(checkSampleEQ)

minNeocortex <- min(checkSampleNeocortex)

minHippocampus <- min(checkSampleHippocampus)

minCerebellum <- min(checkSampleCerebellum)

minStriatum <- min(checkSampleStriatum)

minMOB <- min(checkSampleMOB)

minAllAreas <- min(

minEQ,

minNeocortex,

minHippocampus,

minCerebellum,

minStriatum,

minMOB

)

#Max values

maxFruit <- max(checkSampleFruit)

maxLeaf <- max(checkSampleLeaf)

maxRange <- max(checkSampleRange)

maxBrain <- max(checkSampleBrain)

maxEQ <- max(checkSampleEQ)

maxNeocortex <- max(checkSampleNeocortex)

maxHippocampus <- max(checkSampleHippocampus)

maxCerebellum <- max(checkSampleCerebellum)

maxStriatum <- max(checkSampleStriatum)

maxMOB <- max(checkSampleMOB)

maxAllAreas <- max(

maxEQ,

maxNeocortex,

maxHippocampus,

maxCerebellum,

maxStriatum,

maxMOB

)

```

# Results

| Recent tools have been developed to infer the effect of species interactions on trait evolution, either by modelling trait divergence in co-occurring species from a same guild (e.g. dietary guild; Matching Competition: MC models) or considering that the evolutionary rate depends on the number of co-occurring (?) lineages within the guild (density dependence; linear: DD$\_{lin}$ or exponential: DD$\_{exp}$; @drury2016estimating). After reconstructing primate biogeography history when considering `r length(areaName)` biogeographic areas following [@kamilar2009environmental] based on `r minRange` primate species (@matzke2013probabilistic; @matzke2016stochastic; Figure \@ref(fig:figmap)) as well as primate diet evolution based on `r minFruit + minLeaf` to `r maxFruit + maxLeaf` species (frugivory vs. folivory; see [Dietary guild]), we calculated the likelihoods of models considering the role of species interactions (including competitive scenarios) in the evolution of either the whole brain (using the encephalic quotient, EQ, as a proxy for `r minEQ` to `r maxEQ` frugivorous), or the relative size of specific brain areas associated with foraging-related information perception, processing, or retention (Figure. \@ref(fig:figbrain); comprising `r minAllAreas` to `r max(maxNeocortex,maxHippocampus,maxCerebellum,maxStriatum,maxMOB)` frugivorous species). The use of specific region size relatively to the body mass (see @deaner2000comparative for further consideration of scaling methods), and not raw size, rather depicts the evolutionary evolution of cognitive abilities in terms of allocation rather than abilities per se (although it is vividly discussed whether raw measures are anything informative on "abilities" too @logan2018beyond). We also estimated the likelihoods of simpler models assuming no effect of species interactions, like the Brownian Motion (BM), the Ornstein-Uhlenbeck process (OU) assuming that traits are constrained around on optimal value (e.g. stabilizing selection; see @blomberg2020beyond for a review on these approaches) or the Early-Burst model (EB, @blomberg2003testing), this latter allowing to check for a time-dependence of the evolutionary rate, hence emphasizing that, if any, the density effect is not an artefact due to time dependence. Support for each model was evaluated using an information-theoretic framework [@burnham2002model] based on the weight of Akaike Information Criterion corrected for small samples (AICc) when considering all six models (MC, DD$\_{lin}$, DD$\_{exp}$, BM, OU, EB, see [Models of trait evolution: does interspecific interactions shape brain size evolution?]). Non-competitive models were the most likely in describing the evolutionary history of the EQ, the Neocortex and the Cerebellum (Figure \@ref(fig:figbrain) and \@ref(fig:figresultsevolution)), two areas specifically involved in movement and/or immediate information processing [@wiltgen2004new; @koziol2014consensus; @sokolov2017cerebellum] but also in memory consolidation for the Neocortex [@wiltgen2004new]. By contrast, competitive models were most supported in areas involved in sensory abilities (the main olfactory bulb, MOB), in short-term working memory and long-term spatio-temporal information retention (Hippocampus, @burgess2002human), and in information processing during social interaction (Striatum; @izuma2008processing) (Figure \@ref(fig:figbrain) and \@ref(fig:figresultsevolution)). When density-dependent models were the best fit, the rate (\*r\*, Figure \@ref(fig:figresultsevolution)) suggested an acceleration of the evolutionary tempo together with increased lineage diversity for the Hippocampus and the Striatum, but a slowdown for the MOB.

```{r samplecooccurrence}

minNspeciesCoocc <- round(min(dataRangePrimate$Number\_species\_cooccurrence, na.rm=TRUE), digit=2)

maxNspeciesCoocc <- round(max(dataRangePrimate$Number\_species\_cooccurrence, na.rm=TRUE), digit=2)

meanNspeciesCoocc <- round(mean(dataRangePrimate$Number\_species\_cooccurrence, na.rm=TRUE), digit=2)

seNspeciesCoocc <- round(sd(dataRangePrimate$Number\_species\_cooccurrence, na.rm=TRUE)/sqrt(length(dataRangePrimate$Number\_species\_cooccurrence[!is.na(dataRangePrimate$Number\_species\_cooccurrence)])), digit=2)

whichSpeciesMinNCoocc <- paste(sort(dataRangePrimate$Species[which(round(dataRangePrimate$Number\_species\_cooccurrence, digit=2)==minNspeciesCoocc)]), collapse=", ")

whichSpeciesMaxNCoocc <- paste(sort(dataRangePrimate$Species[which(round(dataRangePrimate$Number\_species\_cooccurrence, digit=2)==maxNspeciesCoocc)]), collapse=", ")

whichSpeciesMinNCoocc <- gsub("\_", " ", whichSpeciesMinNCoocc)

whichSpeciesMaxNCoocc <- gsub("\_", " ", whichSpeciesMaxNCoocc)

minOverlapSpecies <- round(min(dataRangePrimate$Overlap\_average, na.rm=TRUE), digit=2)

maxOverlapSpecies <- round(max(dataRangePrimate$Overlap\_average, na.rm=TRUE), digit=2)

meanOverlap <- round(mean(dataRangePrimate$Overlap\_average, na.rm=TRUE), digit=2)

seOverlap <- round(sd(dataRangePrimate$Overlap\_average, na.rm=TRUE)/sqrt(length(dataRangePrimate$Overlap\_average[!is.na(dataRangePrimate$Overlap\_average)])), digit=2)

whichSpeciesMinOverlap <- paste(dataRangePrimate$Species[which(round(dataRangePrimate$Overlap\_average, digit=2)==minOverlapSpecies)], collapse=", ")

whichSpeciesMaxOverlap <- paste(dataRangePrimate$Species[which(round(dataRangePrimate$Overlap\_average, digit=2)==maxOverlapSpecies)], collapse=", ")

whichSpeciesMinOverlap <- gsub("\_", " ", whichSpeciesMinOverlap)

whichSpeciesMaxOverlap <- gsub("\_", " ", whichSpeciesMaxOverlap)

```

```{r estimateRegGradient, error=TRUE, include=TRUE}

#Hippocampus

model <- get(paste("modelBrain", traitName[3], sep="\_"))

estimateGradientHippocampus <- textEstOneVar(summary(model)$coefficients[2,c(1,4,5)])

testGradientHippocampus <- textTestOneVar(summary(model)$coefficients[2,c(3,6)], statistics="")

#Striatum

model <- get(paste("modelBrain", traitName[6], sep="\_"))

estimateGradientStriatum <- textEstOneVar(summary(model)$coefficients[2,c(1,4,5)])

testGradientStriatum <- textTestOneVar(summary(model)$coefficients[2,c(3,6)], statistics="")

#MOB

model <- get(paste("modelBrain", traitName[7], sep="\_"))

estimateGradientMOB <- textEstOneVar(summary(model)$coefficients[2,c(1,4,5)])

testGradientMOB <- textTestOneVar(summary(model)$coefficients[2,c(3,6)], statistics="")

```

| Next, to understand the directionality of the selection gradient shaped by co-occurrence (i.e. selection for “bigger” or “smaller” brain if the more species), we fitted phylogenetic regressions (see [Phylogenetic regressions] a)). For these linear regressions, the predicted variable was the relative brain size values of the different areas. We considered the average surface of the frugivorous species range that was overlapped by other sympatric frugivorous species, as well as the number of such sympatric frugivorous species across their entire distribution range as covariates. On average ($\pm$ SE), the considered primate species had `r meanOverlap\*100`% of their range overlapping with other species ($\pm$ `r seOverlap\*100`). That ranged from `r minOverlapSpecies\*100`% of overlap (\*`r whichSpeciesMinOverlap`\*), to `r maxOverlapSpecies\*100`% of overlap (\*`r whichSpeciesMaxOverlap`\*). In term of distribution rangethecooccurred on average with other primate species, ingThe number of sympatric species never influenced significantly the relative size of the brain or other specific areas (Table \@ref(tab:tableRegGradient)). Conversely, we found that the percentage of range shared with other species correlated with the relative size of areas which evolutionary history was better described with competitive models: the Hippocampus, the MOB, and the Striatum (Hippocampus: $t$`r testGradientHippocampus`; MOB: $t$`r testGradientMOB`; Striatum: $t$`r testGradientStriatum`). The correlations were all negative (Hippocampus: `r estimateGradientHippocampus`; MOB: `r estimateGradientMOB`; Striatum: `r estimateGradientStriatum`), which means that higher overlap rhymes with lower relative size, insensitive to data and the phylogeny variability (Table \@ref(tab:tabledfsensitivity)). Thus, it suggests that sympatric species are subject to less stringent selection on advanced cognitive abilities.

```{r estimateRegDiversification, error=TRUE, include=TRUE}

#print(traitName)

# increaseHippocampus <- round(exp(summary(model)$coefficients[2,c(1)]), digit=2)

# averageDivRateHippocampus <- round(exp(summary(model)$coefficients[1,c(1)]), digit=2)

# estimateHippocampus <- textEstOneVar(summary(model)$coefficients[2,c(1,4,5)])

# testHippocampus <- textTestOneVar(summary(model)$coefficients[2,c(3,6)], statistics="")

#

# #Neocortex

# model <- get(paste("modelBrainDiversification", traitName[4], sep="\_"))

#

# increaseNeocortex <- round(exp(summary(model)$coefficients[2,c(1)]), digit=2)

# averageDivRateNeocortex <- round(exp(summary(model)$coefficients[1,c(1)]), digit=2)

# estimateNeocortex <- textEstOneVar(summary(model)$coefficients[2,c(1,4,5)])

# testNeocortex <- textTestOneVar(summary(model)$coefficients[2,c(3,6)], statistics="")

#

# #Striatum

# model <- get(paste("modelBrainDiversification", traitName[6], sep="\_"))

#

# increaseStriatum <- round(exp(summary(model)$coefficients[2,c(1)]), digit=2)

# averageDivRateStriatum <- round(exp(summary(model)$coefficients[1,c(1)]), digit=2)

# estimateStriatum <- textEstOneVar(summary(model)$coefficients[2,c(1,4,5)])

# testStriatum <- textTestOneVar(summary(model)$coefficients[2,c(3,6)], statistics="")

#print(c(increaseStriatum, averageDivRateStriatum, estimateStriatum, testStriatum))

```

```{r determineBoomTime}

###-----

## Diversification evolution over time

fraction.v <- c(60, 70, 80, 90, 95)

table\_MAPS\_rates\_time <- c()

for (f in fraction.v){

table\_MAPS\_rates\_time\_transitory <- read.table(paste0("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Scripts/Analysis3\_diversification/diversification/MAPS\_speciation\_rates\_trought\_time\_ClaDS2\_tree\_primate\_complete\_f",f,".csv"),sep=";",header=T)###Hahah nice English Benoit, trought ;)

table\_MAPS\_rates\_time\_transitory$f <- f

table\_MAPS\_rates\_time <- rbind(table\_MAPS\_rates\_time, table\_MAPS\_rates\_time\_transitory)

}

aggregatedSpeciationTime.mean <- aggregate(table\_MAPS\_rates\_time$speciation\_rates, by = list(table\_MAPS\_rates\_time$time), FUN = mean)

aggregatedSpeciationTime.sd <- aggregate(table\_MAPS\_rates\_time$speciation\_rates, by = list(table\_MAPS\_rates\_time$time), FUN = sd)

colnames(aggregatedSpeciationTime.mean) <- c("Time", "Diversification")

colnames(aggregatedSpeciationTime.sd) <- c("Time", "Diversification")

bp.resp <- breakpoints(Diversification ~ Time, data=aggregatedSpeciationTime.mean, breaks=2)

ci.resp <- confint(bp.resp)

dates <- as.data.frame(breakdates(ci.resp))

dateFirstRupt <- as.numeric(round((1-dates[1,])\*(max(aggregatedSpeciationTime.mean$Time) - min(aggregatedSpeciationTime.mean$Time)), digit=2))

dateSecondRupt <- as.numeric(round((1-dates[2,])\*(max(aggregatedSpeciationTime.mean$Time) - min(aggregatedSpeciationTime.mean$Time)), digit=2))

```

| Finally, we investigated the evolutionary consequences of brain evolution by evaluating whether brain sizes were correlated with diversification rates (i.e. speciation minus extinction rates), by using lineage-specific birth-death models of species diversification [@maliet2019model]. Overall, species diversification, estimated based on molecular phylogeny, increased over time (Figure \@ref(fig:figdiversificationTime), particularly in the early and late Miocene, around `r dateFirstRupt[2]` (CI95% = `r paste("[",dateFirstRupt[3],",",dateFirstRupt[1],"]", sep="")`) and `r dateSecondRupt[2]` (CI95% = `r paste("[",dateSecondRupt[3],",",dateSecondRupt[1],"]", sep="")`) Myr ago (Figure \@ref(fig:figdiversificationTime)). Visual inspection clearly suggested positive relationship between diversification rate and the size of brain areas (Figure \@ref(fig:figRegressionDiversification)). Yet, accounting for phylogenetic dependence erased such pattern: In fitted Bayesian regressions, the size of brain size was never significantly associated with an increase in diversification rate (Table \@ref(tab:tableRegDiversification); see robustness in Table \@ref(tab:tabledfsensitivity2)).

# Discussion

[je rappelerais ici les principaux résultats non ? ]

| The use of brain size as a proxy for cognition is a central debate with no optimal solution (see grounded criticism from @deaner2000comparative; @healy2007critique; @logan2018beyond). The current flourishment of consortia, allowing for much more detailed and standardized anatomical measurements (e.g. in primates: @milham2018open), or with standardized behaviourally explicit comparisons (e.g. on captive [@many2019establishing] or wild [@janmaat2021using] primates), might alleviate biases stemming from brain size analysis, but this will take time. In the meanwhile, brain size is a proxy much appreciated in practice, because of its easy accessibility for a “large” number of species. Further, biases might be limited by considering measurement variability [@logan2018beyond] or the mosaic structure of the brain [@barton2000mosaic; @decasien2019primate]. We did both. Although it existed a variability in the data (phylogenetic and on traits), results were robust. In addition, we saw that the evolutionary history between specific brain regions did not equally depend on the number of lineages living in sympatry. The effect of between-species interaction was indeed only evidenced for specific areas, more particularly those involved in immediate information processing based on senses (Main Olfactory Bulb, MOB), in a working memory or in a long-term memory of spatio-temporal information (Hippocampus) and in processing social cues (Striatum). These areas thus imply individual-based and social-based information processing, pinpointing that the two components might be under selection in primates. This supports the general discussion on the importance of social vs. ecological factors to explain primate cognition evolution and diversity [@decasien2017primate; @powell2017re]. Using a modelling approach including metabolic, life-history and game theories, @gonzalez2018inference emphasized that ecological challenges were preponderant (equating around 60% of challenges faced) to explain current human brain size, which then was also substantially promoted by the occurrence of social challenges (equating around 30% of challenges faced). Here, we highlighted that the cognitive function allowing processing sociological or ecological cues are both affected by species sympatry.

```{r calculationValueDiscussion}

#Import data

summaryDataForPlot <- read\_delim("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/OutputEvolModel/Dataplot.txt","\t", escape\_double = FALSE, trim\_ws = TRUE)

summaryDataForPlot <- summaryDataForPlot[!is.na(summaryDataForPlot$geographicCode),]

# summaryDataForPlot <- summaryDataForPlot[summaryDataForPlot$DietaryGuild=="Fruit",]

summaryDataForPlot$Family <- Data\_powell2$MSW05\_Family[match(summaryDataForPlot$Species\_abbrv,Data\_powell2$Species\_abbrv)]#summaryData$Family[match(summaryDataForPlot$Species\_abbrv, summaryData$Species\_abbrv)]

summaryDataForPlot[is.na(summaryDataForPlot$Family ),c(1,2,3)]

#summaryDataForPlot$SpeciesForPhylogeny[which(is.na(summaryDataForPlot$Family))]

summaryDataForPlot$Family[summaryDataForPlot$Species\_abbrv=="Galagoides\_demi"]<-"Galagonidae"

summaryDataForPlot$Family[summaryDataForPlot$Species\_abbrv=="Macaca\_munz"]<-"Cercopithecidae"

summaryDataForPlot$Family[summaryDataForPlot$Species\_abbrv=="Microcebus\_mitt"]<-"Cheirogaleidae"

summaryDataForPlot$Family[summaryDataForPlot$Species\_abbrv=="Mirza\_zaza"]<-"Lorisidae"

summaryDataForPlot$Family[summaryDataForPlot$Species\_abbrv=="Nycticebus\_java"]<-"Galagonidae"

summaryDataForPlot$Family[summaryDataForPlot$Species\_abbrv=="Tarsius\_lari"]<-"Tarsiidae"

lemuriformes.v <-

c(

"Daubentoniidae",

"Lemuridae",

"Indriidae",

"Cheirogaleidae",

"Lepilemuridae"

)

platyrrhini.v <- c(

"Atelidae",

"Aotidae",

"Pitheciidae",

"Cebidae"

)

meanseMOBlemu <- round(c(

mean(

(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %in% lemuriformes.v],

na.rm=TRUE),

sd(

(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %in% lemuriformes.v],

na.rm=TRUE)/

sqrt(

length(

(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %in% lemuriformes.v &

!is.na(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)]))

), digit=2)

meanseMOBother <- round(c(

mean(

(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %nin% lemuriformes.v],

na.rm=TRUE),

sd(

(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %nin% lemuriformes.v],

na.rm=TRUE)/

sqrt(

length(

(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %nin% lemuriformes.v &

!is.na(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass)]))

), digit=2)

meanseStriatumplaty <- round(c(

mean(

(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %in% platyrrhini.v],

na.rm=TRUE),

sd(

(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %in% platyrrhini.v],

na.rm=TRUE)/

sqrt(

length(

(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %in% platyrrhini.v &

!is.na(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)]))

), digit=2)

meanseStriatumother <- round(c(

mean(

(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %nin% platyrrhini.v],

na.rm=TRUE),

sd(

(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %nin% platyrrhini.v],

na.rm=TRUE)/

sqrt(

length(

(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)[summaryDataForPlot$Family %nin% platyrrhini.v &

!is.na(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass)]))

), digit=2)

# summaryDataForPlot[order(summaryDataForPlot$MOB/summaryDataForPlot$Bodymass, decreasing = TRUE),]

# summaryDataForPlot[order(summaryDataForPlot$Hippocampus/summaryDataForPlot$Bodymass, decreasing = TRUE),]

# summaryDataForPlot[order(summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass, decreasing = TRUE),]

```

<!-- From small to long range ( < few km) primates might rely on sensory abilities to decide where to go [@dominy2001sensory]. They are known as visual mammals. In particular, frugivorous species highly benefit from a trichromatic vision that enables them to distinguish fruits within the dense foliage [@osorio1996colour]. Yet, in tropical forests, visual range is easily impeded, being barely greater than a hundred of meters. On the contrary olfactory cues are of longer distance. -->

| Although primates are microsmatic species and better known as visual foragers [@dominy2001sensory; @osorio1996colour], frugivorous species also benefit from olfactory cues processing. Fruits can be highly odorous: The produced ethanol and other chemical compounds can be smelled so as to identify fruit ripeness, but also the location of fruit trees with ripe fruit, although current evidence for this latter case is weak [@nevo2015led]. The Lemuriformes, that are known to prioritize smell compared to other primate species, indeed have the largest relative MOB size (i.e. pondered by body size) in our data (Lemuriformes: mean $\pm$ SE = `r meanseMOBlemu[1]` $\pm$ `r meanseMOBlemu[2]`, other: `r meanseMOBother[1]` $\pm$ `r meanseMOBother[2]`, \@ref(fig:figbrain)). When worthy targets are out of the perceptual range, primates might rely on internal memories of the resource distribution in space, and of their availability period, to forage efficiently [@janson2007wild; @trapanese2019review]. In this system, the Hippocampus occupies a key position: It hosts (inter)neurons that encode for spatial location and orientation (known as place, grid or head cells, @grieves2017representation) and is home of episodic memory [@ranganath2016hippocampus]. In addition, when foraging, environmental cues might be complemented by social cues, which processing can involve the Striatum. Platyrrhini, and callitrichine in particular, are known to form poly-specific associations [@heymann2000behavioural] and indeed show the highest relative size of the Striatum in our data (Platyrrhini: mean $\pm$ SE = `r meanseStriatumplaty[1]` $\pm$ `r meanseStriatumplaty[2]`, other: `r meanseStriatumother[1]` $\pm$ `r meanseStriatumother[2]`, \@ref(fig:figbrain)). It has been shown that individuals tend to use social or environmental cues depending on their reliability [@rafacz2003environmental; @dunlap2016foraging]. A lesser scrutinized function of the Striatum is also that of supporting goal-directed behaviour and planning abilities [@johnson2007integrating]. Overall, we expect the size of brain areas involved in dissecting socio-environmental cues to be under strong positive selection [@decasien2017primate; @barton2006olfactory]. Here, on the contrary, we showed that species co-occurrence acts as a brake to such positive selection since the size of these areas were negatively associated with species co-occurrence. This was the result of a slowdown of the evolutionary rate for the MOB, but an acceleration (thus towards lower size) of the evolutionary rate for the Hippocampus and Striatum.

| Competition is generally the first-thought mechanism to describe community structures [e.g. @rocha2015role]. Following the principle of an arms race between species (Red Queen scenario, @van1973new), it would have been logical to see species co-occurrence as an additional positive driver towards increase size. A multi-species case stands yet as a peculiar situation. In particular, inter-species site exclusion in primates has been observed only in gibbons [@suwanvecho2012interspecific]. Thus, given that primates restrain their space-use to a limited area, their home range, they will suffer from more intense depletion (and consequently unpredictability), of their environment. If this latter is too important, this could alleviate the benefices purported by foraging cognition [@grove2013evolution; @robirainreview]: The environment would be too complex to read through it, and conspecifics might be thus be too error-prone to rely on them. Positive selection for "bigger" areas supporting foraging efficiency would be relaxed, and, given the functioning and maintenance cost of the brain [@raichle2006brain], this could even turn into a selection for "smaller" sizes of brain areas related to socio-ecological cue processing.

| Yet, as much as social species could exploit cues provided by conspecifics, a species might also benefit from using cues of other species. To settle to new coral reefs, fishes use pops and clicks of other fishes as an honest signal for resourcefulness there [@gordon2019acoustic], mangabeys follow calls from hornbills to locate fruiting trees [@olupot1998fruit], and interactions even happen across kingdoms, with migratory birds interpreting phenological cues as synonymous of insect availability [@mcgrath2009flower]. These signals should not involve true social reading, thus should be processed by areas such as the Neocortex which process such sensory cues [@barton1996neocortex]. This could explain why the size of this area was actually better described by evolutionary models with stabilizing selection and did not follow the pace of the three aforementioned areas. Despite a potential increase load of stimuli (due to the cues provided by other species), the Neocortex size however did not correlate positively with sympatry rate. Perhaps the inter-specific cues do not add, but simply replace other used cues. In addition, given that areas affected by sympatry are far smaller than those that are not, there is no surprise that the deficit in allocation to these areas, potentially to the benefits of the Neocortex, is not observed through an increase of the Neocortex size.

| Finally, we observed that primate diversification rate increased along time particularly around -`r dateFirstRupt[2]` and -`r dateSecondRupt[2]` Myr. This corroborates previous findings about diversity boom in primate lineages early and late Miocene as a consequence of a sharp decrease in extinction rate [@arbour2017major; @springer2012macroevolutionary] due to the emergence of more favourable environmental conditions stemming from a progressive warming after harsh temperature cooling that started earlier in the Oligocene until reaching a mid-Miocene Climatic Optimum [@fleagle2006biogeography]. Given the observed effect of species co-occurrence on brain size selection trends, species living in areas with or without competitive species would thus not be under the same selective regime. This would explain why we did not observe a link between the size of brain areas negatively affected by sympatry and their evolutionary success, approximated by their diversification rate. Why, nonetheless, the whole brain size is not correlated to diversification rate, while it is unaffected by the density of sympatric species, remains puzzling given that higher cognitive abilities are associated with higher ecological success since they act as a "cognitive buffer" to environmental challenges [@sol2007big]. For these reasons, larger brain size is indeed associated with higher diversification in birds [@sayol2019larger]. To sum up, these results suggest that the encephalisation boom observed in primates shall not be explained by a global, or area-restricted, encephalization increase, as suggested for Hominins [@melchionna2020macroevolutionary].

# Conclusion

In the end, the inter-specific effect on cognition was here mainly viewed under the prism of foraging and was limited to within primates. Without further evidence, it is as likely to hold if considering all potential competitors, that is not limited to an arbitrary taxa (see evidence of primate and non-primate interactions, reviewed in @heymann2015unlike), and in other contexts, such as the social environment. In fact, the general hypotheses on cognition evolution, discussed within species, could be broadened to a between-species context: polyspecific social associations do exist [@porter2001benefits], as well as inter-species territory defense [@drury2020competition; @losin2016ecological] or imitation and copying [@persson2018spontaneous; @pepperberg2002allospecific]. As Alice said “'It's a great huge game of chess that's being played—all over the world” (@Carroll, Chapter II) and all individuals are just pieces to play with or against, no matter the species.

# Methods

Data processing, analyses, and plots were computed with R software version 4.0.3 [@Rsoftware]. Used codes and data are freely available at [https://github.com/benjaminrobira/Temporal\_memory\_and\_foraging\_efficiency](https://github.com/benjaminrobira/Temporal\_memory\_and\_foraging\_efficiency).

## Data Collection

Note that in all these analyses, we discarded \*Homo sapiens\* and \*Macaca sylvanus\*, this latter being too geographically isolated. A summary of available data per species is presented in Appendix Figure \@ref(fig:figmap).

### Phylogeny

We used a block of chronogram trees of the primate taxon of the 10kTrees project (downloaded on the 11/05/2021, version 3), as well as a consensus tree of 1000 trees for the subsequent phylogenetic analyses. The trees contain `r length(phylo\_init$tip.label)` primate species.

### Brain data

Brain data were obtained from @decasien2019primate for whole brain and all mentioned other parts (Cerebellum, Hippocampus, Main Olfactory Bulb (MOB), Neocortex, Striatum), @powell2017re and @powell2019maternal for whole brain, Cerebellum and Neocortex size, @todorov2019primate for Hippocampus and Neocortex size, @grueter2015home for the whole brain size and @navarrete2018primate for the whole brain, Cerebellum, Hippocampus and Striatum size. They were freely available in the main manuscript or supplementary materials. When the species was represented multiple times within dataset, we obtained a unique attribute by averaging it. From the global endocranial brain volume, we obtained the Encephalization Quotient (EQ, N$\_{EQ,max}$ = `r maxEQ`) as follows [@decasien2017primate]

\hfill

\begin{center}

$\mathrm{EQ}=1.036 \times \mathrm{Brainvolume}/ (0.085 \times \mathrm{Body mass}^{0.775})$

\end{center}

\hfill

with the brain volume in cm$^{3}$, 1.036 g/cm$^{3}$ being the assumed homogeneous brain density, and the body mass in g. EQ indicates whether the brain size ranges above (> 1) or below (< 1) expected given the body mass. Body mass was obtained from @decasien2017primate, @powell2017re, @grueter2015home and @pearce2013space.

The sub-parts of the brain were chosen because they were involved in immediate sensory information processing (MOB, N$\_{MOB,max}$ = `r maxMOB`), in movement and/or associate information processing and retention (Neocortex, N$\_{Neocortex,max}$ = `r maxNeocortex`, @wiltgen2004new; Cerebellum, N$\_{Cerebellum,max}$ = `r maxCerebellum`, @koziol2014consensus; @sokolov2017cerebellum), short-term working memory and long-term spatio-temporal memory (Hippocampus, N$\_{Hippocampus,max}$ = `r maxHippocampus`, @burgess2002human). The Striatum (N$\_{Striatum,max}$ = `r maxStriatum`), which supports information processing during social interaction (i.e. social reward assessment; @izuma2008processing), was chosen so as to serve as a comparative “null” area. To investigate their evolutionary history, we used the ratio between their volume and

<!-- that of the whole brain. -->

### Diet and body mass data

Percentage of frugivory and/or folivory was obtained based on freely available dataset from @decasien2017primate and @powell2017re for the frugivory and folivory rate, or @willems2013collective for the folivory rate. Body mass data were available from @decasien2017primate, @powell2017re, @grueter2015home and @pearce2013space.

### Ranging Data

Current geographic (maximal possible) range of each primate species was assessed using ranging maps provided by the IUCN red list [@IUCN]. Ranging data were available for `r nrow(matrixRangingSensitivity[!is.na(matrixRangingSensitivity[,1]),])` species among the `r length(phylo\_init$tip.label)` represented in the 10kTrees primate phylogeny.

## Primate species co-occurrence

One to multiple large-scale geographic areas were assigned to each species as soon as the species current range overlapped in surface at `r geographicThresholdVector[1]\*100` (low threshold) or `r geographicThresholdVector[2]\*100`% (high threshold; the maximum was chosen to `r geographicThresholdVector[2]\*100`% because on present data, a species could occupy as far as three areas; Figure \@ref(fig:figmap)). Overlap was calculated with the “gIntersection” function from the \*rgeos\* package [@rgeos] applied to Mercator-projected data to get the overlap contour, and the “area” function from the \*geosphere\* package [@geosphere], applied directly on unprojected longitudinal-latitudinal data for area size calculation. These geographic areas were initially, manually delimited using Google earth professional (v7.3.3) as a combination of the environment topology and geographic regionalization relative to the primate taxonomy [@kamilar2009environmental]. Based on the structure (i.e. number of species and their phylogenetic relationship) of primate communities at different field sites, @kamilar2009environmental determined clusters of sites with highly similar community structures that were shaped by both the environment geography and climatic correlates. The considered geographic areas are represented in Figure \@ref(fig:figmap). The chosen scale for the areas is large because (1) retracing history of a large number of areas necessitates considerable computational means. In addition, this drastically increases computational time of phylogenetic model of brain trait evolution too. Furthermore (2), all species and particularly primate species suffer(ed) from recent extinction [@pavoine2019mammal], with reduction of ranging areas at an unpreceeding speed rate. Finer geographic characterization would therefore give too much weight to such anthropogenic effect that recently altered species distribution (e.g. evidenced on the North American fauna in @pineda2021mammal). Finally, note that the north part of Africa and the south of Europe were discarded despite the presence of one primate species (\*Macaca sylvanus\*), because of its geographical complete isolation and repeated intervention of human people in population maintenance [@modolo2005phylogeography]. Hence, \*Macaca sylvanus\* is not considered in this study.

| We retraced the history of the lineage ranges based on current observations of species range using the \*BioGeoBEARS\* package [@matzke2013probabilistic] following the biogeographic stochastic mapping algorithm [@matzke2016stochastic]. This algorithm aims to fit, among others, non-time-stratified dispersal-extinction-cladogenesis (DEC) models (used here), specifically suiting analyses of range data since it accounts for spatially explicit processes of cladogenetic and anagenetic events (see @matzke2013probabilistic for further details on these events). To reconstruct the evolution of species range, we fixed the maximum numbers of areas that could be occupied by a lineage at one time to three areas. A too high number of areas that can be occupied simultaneously drastically increases computational time. Here, we therefore chose that a species can at most occupy three areas since it offers the possibility to occupy a complete mainland continent. Finally, because these history reconstructions are likely to vary, for each run of DEC models, we obtained `r numberSimulations` stochastic maps that were all used in subsequent phylogenetic model fitting (see [Phylogenetic models]) to account for uncertainty of these ancestral range estimations (see [Models of trait evolution: does interspecific interactions shape brain size evolution?] (b)).

## Dietary guild

We classified species as either “frugivorous” or “folivorous” based on the availability of frugivorous rate and folivorous rate, prioritizing fruvigory over folivory. First, a species would be classified as frugivorous if the frugivory rate was at least above `r frugivoryThresholdVector[1]` (low threshold) or `r frugivoryThresholdVector[2]`% (high threshold). If this was not the case, or frugivory rate was unavailable, a species could be classified as folivorous if the folivory rate was at least above `r folivoryThresholdVector[1]` (low threshold) or `r folivoryThresholdVector[2]`% (high threshold). Otherwise, @decasien2017primate gave a binary classification of diet, species being categorized as frugivorous or folivorous, partly based on anatomical criteria. Whenever the rate was not available, we referred to this classification. In any other cases, the species was discarded.

```{r transitionMatrix}

transitionMatrix <- matrix(NA, nrow=repetition, ncol=2)

for(a in 1:2){

for(b in 1:2){

for(c in 1:2){

for(d in 1:10){

start=which(is.na(transitionMatrix[,1]))[1]

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_simmap\_transition",a,"\_",b,"\_",c,"\_",d, ".txt", sep=""))

transitionMatrix[start,1] <- toAdd[1,1]

transitionMatrix[start,2] <- toAdd[2,1]

}, error=function(e){

#Do nothing

}

)

}

}

}

}

minProba.v <- apply(abs(transitionMatrix), 2, min)

maxProba.v <- apply(abs(transitionMatrix), 2, max)

```

| Frugivory rate was prioritized over folivory because we considered that since fruits are a highly palatable food source, it would be the key item that drives the foraging strategy (and associate consequence on brain selection), even if less consumed. Additionally, to consider frugivory, we used a lower rate than for folivory for two reasons. First, such static rate does not reflect potential seasonality in fruit eating [e.g. @masi2009western], which is generally shorter, hence a lower overall frugivory rate. Second, frugivory rate is likely to be underestimated in part because primates generally spend more time feeding on leaves than fruits, while rates are often based on relative feeding time, or observation frequency at the individual or group unit of feeding event. Finally, the methodology to obtain this rate could additionally vary (e.g. in addition to the two aforementioned estimations, one could also rely on the proportion of species targeted for their fruits/leaves). For all these reasons, we used two threshold levels (low, `r frugivoryThresholdVector[1]`%, or high, `r frugivoryThresholdVector[2]`%) to classify a species as frugivorous, as well as two threshold levels (low, `r folivoryThresholdVector[1]`%, or high, `r folivoryThresholdVector[2]`%) to classify a species as folivorous.

Considering diet as a binary variable (frugivory versus folivory), we retraced the evolutionary history of such discrete traits based on a continuous Markovian process (extended Mk models) and relying on a Bayesian approach [@bollback2006simmap], using the “simmap” function of the \*phytools\* package [@phytools] and internally estimating the prior probability of trait (i.e. at the root) but with no prior on the transition matrix. Again, the obtained character history is in no case certain. Therefore, for each run, we obtained `r numberSimulations` stochastic character maps that were used in subsequent phylogenetic model fitting [Phylogenetic models] to account for uncertainty of these ancestral diet estimations (see [Phylogenetic models, Models of trait evolution: does interspecific interactions shape brain size evolution?] (b)).

## Phylogenetic models

### Models of trait evolution: does interspecific interactions shape brain size evolution?

\hfill

(a) Fitting models of trait evolution

\hfill

We focused on frugivorous primates, because sample size was otherwise insufficient, and fitted phylogenetic model of EQ - or relative size of a specific brain area – evolution with and without species competitions. Models were fitted on different sample sizes due to non-availability of some data. Specifically, models using EQ included `r minEQ` to `r maxEQ` frugivorous species. Other models included more reduced sample sizes (in species number): Striatum (`r minStriatum` to `r maxStriatum`), MOB (`r minMOB` to `r maxMOB`), Neocortex (`r minNeocortex` to `r maxNeocortex`), Hippocampus (`r minHippocampus` to `r maxHippocampus`), Cerebellum (`r minCerebellum` to `r maxCerebellum`). Prior fitting, trait parameters were log-transformed to reach more symmetrical distributions. Models without competition, Brownian Motion (i.e. BM), Orstein-Uhlenbeck process (i.e. OU, model with stabilizing selection), or Early-Burst model (i.e. EB, for assessing a time-dependence of the evolutionary rate) were fitted using the “fitContinuous” function from the \*geiger\* package [@geiger3; @geiger5]. Using the evolutionary history of species distribution (see [Primate species co-occurrence]) and of diet (see [Dietary guild]), we fitted competitive models using the “fit\_t\_comp” function from the \*RPANDA\* package [@RPANDA]. These competitive models notably account for interaction matrices that are built on the evolutionary history of species co-occurrence and diet. These interaction matrices retrace, along the phylogenetic tree, which frugivorous lineages were present within the same geographic areas (see @drury2016estimating). We fitted three different competitive models. The matching competition model (MC) may consider divergence of traits of co-occurring lineages from a same dietary guild due to repulsion of traits (character displacement) [@drury2016estimating]. Here, that would mean that co-occurring species would tend to have either lower or higher EQ or relative brain size. Otherwise, we modelled trait evolution accounting for linear (DD$\_{lin}$) or exponential (DD$\_{exp}$) density-dependence [@drury2016estimating; @weir2013diversity]. Density-dependence means that the evolutionary rate $\lambda$ varies either positively or negatively as a function $f$ of the number of co-occurring lineages sharing the same diet such as

\begin{center}

\hfill

$f\_{lin}(\lambda)=\lambda\_{0}(1 + r)$

$f\_{exp}(\lambda)=\lambda\_{0}\exp(rL)$

\hfill

\end{center}

where $\lambda\_{0}$ corresponds to the value of the initial ancestor, $L$ indicates the number of lineages, $r$ allows for modelling the speed and direction of the dependency to lineage number ($r>0$ leads to an increase of trait changes, while $r<0$ leads to a decline of the trait changes).

All these models were repeated `r numberSimulations` times, using `r randomSampling` different combination of the evolutionary history of ranging and diet. They were then compared within an information-theoretic framework [@burnham2002model] based on the weight of Akaike Information Criterion corrected for small samples (AICc) when considering all six models (MC, DD$\_{lin}$, DD$\_{exp}$, BM, OU, EB). The model weight then depicts the probability that it best describes the observed evolutionary pattern among the tested models.

\hfill

(b) Dealing with data uncertainty and parameter sensitivity

\hfill

In this analysis, uncertainty can stem from two sources. First, the true phylogeny is never known with certainty, and is estimated through Bayesian inference, thus we used the consensus tree from the 10kTrees project, which averages the phylogeny among 1000 possible estimated trees.

| Similarly, the estimated evolutionary history of the diet and ranging might vary as well. Second, for each species, trait estimates could vary slightly among datasets (see Appendix Figure \@ref(fig:figvariabilitydata)). Particularly, although correlations seem good enough, it existed a variation in absolute measurement (Appendix Figure \@ref(fig:figvariabilitydata)), while, in order to increase the overall number of species, trait values were not mandatorily from a single dataset. In addition, this study is based on several arbitrary thresholds, namely (i) to assess species co-occurrence (see Appendix Figure \@fig(fig:figcomparison)) and (ii) to assess the species dietary guild (see Appendix Figure \@ref(fig:figvariabilitydata)) which can cause sensitivity of the results to the chosen parameters. To account for these three sources of variability we refitted several times the six models of trait evolution (BM, OU, EB, MC, DD$\_{lin}$ and DD$\_{exp}$) with (1) various biogeography and dietary evolutionary history estimations, (2) random samples of the dietary and brain traits in case of multiple values available (i.e. equal probability for each possible value to be selected) and (3) used the low or high threshold for assessing frugivory, folivory and geographic co-occurrence.

| Eventually, it means that the results for each model represent the average of `r numberSimulations` (uncertainty on diet/ranging evolution) x `r randomSampling` (uncertainty in brain/diet rate data) x `r length(geographicThresholdVector)` (geographic overlap threshold) x `r length(frugivoryThresholdVector)` (frugivory threshold) x `r length(folivoryThresholdVector)` (folivory threshold) = `r numberSimulations\*randomSampling\*length(geographicThresholdVector)\*length(frugivoryThresholdVector)\*length(folivoryThresholdVector)` sub-models.

### Models of species diversification

We investigated how the primate taxon diversified over time. Lineage-specific diversification rates were estimated using an updated version of the \*ClaDS\* algorithm [@maliet2019model] boosted for computational speed based on data augmentation techniques [@maliet2020fast]. Particularly, we used \*ClaDS2\*, the model with constant turnover (i.e. constant ratio between extinction and speciation rates). This Bayesian approach considers speciation rate heterogeneity by modeling small shifts in this rate at speciation events. In other words, the daughter lineage is assumed to inherit new speciation rates that is sampled from a log-normal distribution with an expected mean value $log(\alpha \lambda)$ (where $\lambda$ represents the parental speciation rate and $\alpha$ is a trend parameter), and a standard deviation $\sigma$. Three independent chains were run until their respective convergence was validated by a Gelman-Rubin diagnostic criterion [@gelman1992inference]. The analysis relied on the use of a consensus tree of primate phylogeny from [@dos2018using]. This latter provides a robust phylogenetic tree for 367 primate species (while the 10kTrees primate phylogeny has only `r length(phylo\_init$tip.label)` species).

| Such analysis necessarily depends on a prior estimation of the sample representativeness, that is, the fraction of "known" taxa among all possibles ones. @estrada2017impending estimated that, given current knowledge, the primate lineage should be composed of 504 species. This means that the current sampling fraction is around 73%. We thus parameterized the \*ClaDS\* algorithm with this value for the estimate sampling fraction. Yet, given that the extant number of primate species is subject to controversy, and also because the estimated sampling fraction affects diversification rate estimations, we replicated our analyses with a range of sampling fractions from 95% down to 60%. At the end of each run, we extracted the maximum of the \*a posteriori\* net diversification rate of each primate species, as well as the mean rate (given all lineages) through time.

### Phylogenetic regressions

\hfill

(a) Determining the direction of the selection gradient shaped by interspecific competition

\hfill

To determine the direction of the selection of species co-occurrence on size of brain regions for which competitive models fitted the best, we fitted Gaussian Pagel's lambda phylogenetic regressions (i.e. a derivative of the Brownian Motion model, for which the phylogenetic variance-covariance matrix has all coefficients but its diagonal multiplied by lambda) for each brain region individually and for frugivorous species only. We used the Pagel's lambda model so as to relax the hypothesis of Brownian Motion since we specifically focused on brain areas for which the evolutionary history was best described by competitive models. Here specifically, we considered the least stringent frugivory assessment, with frugivory threshold fixed to `r frugivoryThresholdVector[1]`%, folivory threshold fixed to `r folivoryThresholdVector[1]`%. If, due to data variability, a species did not robustly fit into the categorical classification “frugivorous versus folivorous” (i.e. could be either of the two), it was considered as frugivorous nonetheless.

| The response variable was the relative size of areas shown as better described by competitive phylogenetic scenario (see above). Due to data variability, we took the mean of the possible values given the different datasets, and assessed the sensitivity using non-averaged values (see Model Robustness). In this model, the covariates (i.e. continuous predictors) were the average percent of the range surface overlapping with other sympatric frugivorous species, and the number of frugivorous sympatric species (both were square rooted, to reach symmetrical distribution). For a given species A, sympatry with another species B was considered when species B range overlapped on more than 10% of the range of species A. This was done to reduce noise induced by coarse identification of species range.

\hfill

(b) Diversification analysis

\hfill

In the same way than explained above, we fitted Gaussian Pagel's lambda phylogenetic regressions of the different brain traits against the diversification rate (i.e. accounting for both, speciation and extinction) estimated for each species by the \*ClaDS\* algorithm. Again, we took the mean of the brain trait values for the main model and assessed the sensitivity by re-running the model several times using non-averaged values in this case.

\hfill

(c) Model implementation

\hfill

(i) Direction of the selection gradient shaped by interspecific competition

\hfill

Models were fitted using the “phylolm” function from the \*phylolm\* package [@phylolm], with the lambda parameter (i.e. indicating whether the trait is subject to selection, or corresponds to Brownian Motion, if $\lambda$ tends towards 1) estimated by maximum-likelihood (argument “model” set to “lambda”). Bootstrapping over `r repetitionBootstrap` independent replicates was done so as to obtain confidence intervals. Other function parameters were set to default. Prior fitting, the covariates were square-rooted to reach more symmetrical distribution. Necessary assumptions on the normal distribution of residuals and homoscedasticity were visually assessed and pointed out no violation (see Appendix [Model assumptions]). We did not observe correlation issue among predictors either [max VIF < 2, @mundry2014statistical].

\hfill

(ii) Diversification analysis

\hfill

We could not compute phylogenetic regressions to link diversification and brain traits using a frequentist approach because it lead to violation of homoscedasticity. Instead, we fitted Bayesian phylogenetic regressions using the "MCMCglmm" function of the \*MCMCglmm\* package [@MCMCglmm]. Each chain was based on a burnin period of `r burnin` iterations, among a total of `r nitt` iterations, and was sampled every `r thin` iterations. We used the least informative priors. Fixed priors were let to default (Gaussian distribution of mean 0 and variance $10^{8}$). Prior on random effects and residuals were set to follow an inverse-Wishart distribution with a variance at limit ($V$) of 1, and a degree of belief ($nu$) of 0.02. We checked model convergence by fitting three chains, and calculated the Gelman-Rubin criterion [max value < `r round(max(gelmanRubinValues), digit=2)`; @gelman1992inference], as well as checked autocorrelation (max absolute value < `r round(max(valueAutoCorr), digit=2)`) using the respective "gelman.diag" and "autocorr.diag" functions from the \*coda\* package [@coda]. In Appendix [Model assumptions], we present trace and distributions of posterior estimates. We further checked the quality of the posterior by visually assessing the Q-Q plot of the posterior with that of a Gaussian distribution of mean 0 and sd 1 (see Appendix [Model assumptions]). We present the estimate together with the 95% credibility interval centered on the mode (Highest Density Posterior, HDP), together with a MCMC p-value (pMCMC) that corresponds to the probability that the estimate ($\beta$) is positive if the mean estimate ($\hat{\beta}$) is negative (i.e. $P(\beta>0|\hat{\beta}<0)$), or if the mean estimate is positive, the probability that the estimate is negative (i.e. $P(\beta<0|\hat{\beta}>0)$).

\hfill

(d) Model robustness

\hfill

To assess frequentist model stability with regards to singular points, we computed the DfBetas (variation in estimates) by discarding one observation at a time of the "standard" dataset used to fit the main model, based on the consensus tree.

| To assess the sensitivity to (i) the variability in data and (ii) phylogeny uncertainty, we refitted the models using `r repetitionTrees` phylogenetic trees among the 10000 possible trees from the 10kTrees project. For each of these trees, we fitted the model `r repetitionModels` times, allowing random sampling for data when we had multiple value (e.g. if body mass was provided by different datasets etc.). For the diversification analysis specifically, we also assessed the sensitivity to changes in hypothesized fraction of known primate taxon by refitting the models for values ranging between 60 to 95% (as specified before) using the "standard" dataset and the consensus tree.

| The results of these assessment (min-max of estimates) are shown in Appendix [Model stability]. It emphasizes weak sensitivity of the results.

# Acknowledgements

We considerably value the help provided by Jonathan Drury in making some scripts available, but mostly for helping us in solving issues encountered with the use of functions of his own in the \*RPANDA\* package in \*R\*, and that of Marie-Claude Quidoz for assistance for using the CEFE cluster. We thank Simon Benhamou and Manon Clairbaux for discussion and advices on spatial projections, and M. Quéroué, V. Lauret, A. Caizergues and C. Teplistky for feedback on Bayesian computations too. Finally, this work could not have been possible without prior data collection from the IUCN Red List (primate ranging), the 10kTrees project (phylogenetic trees), and Alexandra R. DeCasien and collaborators, Lauren E. Powell and collaborators, Orlin S. Todorov and collaborators, Erik P. Willems and collaborators, Fiona Pearce and collaborators, Navarrete and collaborators, and Cyril C. Grueter who provided primate trait data we used as (supplementary) material with their articles, as well as Nicholas J. Matzke for available algorithm scripts allowing us to implement and better understand the methods. Their indirect input is therefore tremendous.

| Both authors were supported by a doctoral grant from the \*École Normale Supérieure\*, Paris. BR received support from the \*Centre d’Écologie Fonctionnelle et Évolutive\*, Montpellier and the \*Musée de l'Homme\*, Paris. BP-L received support from the \*École Normale Supérieure\*, Paris and the \*Muséum National d’Histoire Naturelle\*, Paris.

# Authors' contribution

BR conceived the study, collected, cleaned and analyzed the data, drew the figures and wrote the first version of the manuscript and subsequently revised it. BP-L implemented the ClaDS algorithm with our data, helped with other analyses, adapted them for, and ran them on, the cluster, and revised the manuscript multiple times. The authors declare having no conflict of interest. All authors gave final approval for publication and agree to be held accountable for the work performed therein.

\newpage

```{r tableRegGradient, include=TRUE}

knitr::kable(results.df\_gradient, escape=TRUE, booktabs = TRUE, caption = "Model estimates and significance of phylogenetic regressions to assess the selection gradient direction | Est.=Estimate, CI2.5\\%=Lower border of the CI95\\%, CI97.5\\%=Upper border of the CI95\\%, Sd=Standard deviation, t=Statistics t-value. The brain area (as well as the associated sample size) are indicated prior each list of estimates. The transformation applied to variables are indicated between brackets (logarithm, log, or square-root, sqrt), as well as the ponderation by bodymass (/bodymass).") %>%

kableExtra::column\_spec(2:ncol(results.df\_gradient), bold = toPlotBold) %>%

kableExtra::kable\_styling(latex\_options = "striped") %>%

kableExtra::kable\_styling(latex\_options="scale\_down") #%>%

#kableExtra::kable\_styling(latex\_options = "HOLD\_position")

```

```{r}

#Round numbers table

results.df\_diversification[seq(from=4, to=nrow(results.df\_diversification), by=4),c(2,3,4)] <-apply(results.df\_diversification[seq(from=4, to=nrow(results.df\_diversification), by=4),c(2,3,4)], 2, function(x){roundIntelligent(as.numcharac(x), digit=2)})

results.df\_diversification[seq(from=1, to=nrow(results.df\_diversification), by=4),1] <- gsub("\\(log\\) ", "", results.df\_diversification[seq(from=1, to=nrow(results.df\_diversification), by=4),1])

results.df\_diversification[seq(from=1, to=nrow(results.df\_diversification), by=4),1] <- gsub("\\(\\/bodymass, log\\) ", "", results.df\_diversification[seq(from=1, to=nrow(results.df\_diversification), by=4),1])

```

\newpage

```{r tableRegDiversification, include=TRUE}

knitr::kable(results.df\_diversification, escape=TRUE, booktabs = TRUE, caption = "Model estimates and significance of Bayesian phylogenetic regressions to assess the diversification pattern | Est.=Estimate, HDP2.5\\%=Lower border of the 95\\% Highest Posterior Density, HDP97.5\\%=Upper border of the 95\\% Highest Posterior Density, Eff. samp.=Effective sample (adjusted for autocorrelation). The brain area (as well as the associated sample size) are indicated prior each list of estimates. The logarithm transformation was applied to variable and is indicated between brackets (log), as well as the ponderation by bodymass (/bodymass).") %>%

kableExtra::column\_spec(2:ncol(results.df\_diversification), bold = toPlotBoldDiversification) %>%

kableExtra::kable\_styling(latex\_options = "striped") %>%

kableExtra::kable\_styling(latex\_options="scale\_down") #%>%

#kableExtra::kable\_styling(latex\_options = "HOLD\_position")

```

\newpage

```{r prepareMap}

#Reimport areas with cropping

centroid <- matrix(NA, ncol=2, nrow=length(areaName))

for(i in 1:length(areaName)){

areaTransitory <- readOGR(dsn=paste("T:/IUCN\_data\_primate/Geographic\_areas/Shapefiles/",areaName[i],".shp",sep=""))

areaTransitory = clgeo\_Clean(areaTransitory)

areaTransitory <- spTransform(areaTransitory, CRS("+proj=longlat +datum=WGS84 +no\_defs +ellps=WGS84 +towgs84=0,0,0"))

#Have mercator for intersection

areaTransitory <- spTransform(areaTransitory, CRS("+proj=merc +datum=WGS84 +no\_defs +ellps=WGS84 +towgs84=0,0,0"))

areaTransitory <- gIntersection(areaTransitory, worldMap\_mercator, byid=FALSE)

areaTransitory = clgeo\_Clean(areaTransitory)

#Reunite polygon in case

areaTransitory <- gBuffer(areaTransitory, byid=F, width=0)

#back transform to long/lat

areaTransitory <- spTransform(areaTransitory, CRS("+proj=longlat +datum=WGS84 +no\_defs +ellps=WGS84 +towgs84=0,0,0"))

assign(paste("area", i, sep="\_"), areaTransitory)

if(i==1){

centroid[i,] <- c(summary(areaTransitory)$bbox[1,2] + 5, summary(areaTransitory)$bbox[2,1])

}

else if (i==2){

centroid[i,] <- c(summary(areaTransitory)$bbox[1,2] - 5, summary(areaTransitory)$bbox[2,1] - 5)

}

else{

centroid[i,] <- geosphere::centroid(areaTransitory)

}

}

warnings()

```

```{r figmap, include=TRUE, warning = FALSE, message = FALSE, results= 'hide', fig.width=7, fig.height=7, fig.cap=paste("Geographic areas used for ancestral range reconstruction | Depicted is the Natural Earth projection of the world. Areas were defined as a combination of geographic and environmental criteria relatively to the primate taxonomy following results from [@kamilar2009environmental]: (1) East Madagascar (2) West Madagascar (3) West Africa (4) Central Africa (5) East/South Africa (6) Central America (7) North South-America (8) South South-America (9) West Asia (10) Central/East Asia (11) South Asia (12) Asian peninsula and islands. Note that the north part of Africa and the south of Europe were discarded despite the presence of one primate species (\*Macaca sylvanus\*), because of its geographical complete isolation and repeated intervention of human people in population maintenance [@modolo2005phylogeography]. Hence, \*Macaca Sylvanus\* is not considered in this study.", sep="")}

#abbreviation name are available in https://proj.org/operations/projections/index.html

#help for graticule: https://rpsychologist.com/working-with-shapefiles-projections-and-world-maps-in-ggplot

proj.map <- "+proj=natearth +datum=WGS84 +no\_defs +ellps=WGS84 +towgs84=0,0,0"

centroid <- as.data.frame(centroid)

colnames(centroid) <- c("long", "lat")

centroid$nudge\_y <- 0

centroid$nudge\_y[5] <- -5

library(tidyverse)

library(ggspatial)

library(rnaturalearth)

world <- ne\_countries(scale = "medium", returnclass = "sf")

# graticule (Robin)

grat <- readOGR(dsn="C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Formap", layer="ne\_110m\_graticules\_15")

grat\_df <- fortify(grat)

bbox <- readOGR(dsn="C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Formap", layer="ne\_110m\_wgs84\_bounding\_box")

bbox\_df<- fortify(bbox)

bbox.sf <- sfheaders::sf\_polygon( obj = bbox\_df

, x = "long"

, y = "lat"

, polygon\_id = "group"

)

sf::st\_crs(bbox.sf) <- st\_crs(world)

grat\_df <- grat\_df[,c(1,2,6)]

colnames(grat\_df) <- c("x", "y", "linestring\_id")

grat\_df.sf <- sfheaders::sf\_line(obj = grat\_df)

sf::st\_crs(grat\_df.sf) <- st\_crs(world)

#Main map

map <- ggplot() +

geom\_sf(data = bbox.sf, fill=adjustcolor("white", alpha.f=0.45), col="white") +

geom\_sf(data = grat\_df.sf, col="darkgrey", lty=2) +

geom\_sf(data = world, fill = "black", col = "white") +

theme(panel.background=element\_rect(fill="white"),

panel.grid = element\_blank(),

axis.title.x=element\_blank(),

axis.text.x=element\_blank(),

axis.ticks.x=element\_blank(),

axis.title.y=element\_blank(),

axis.text.y=element\_blank(),

axis.ticks.y=element\_blank()

)

#Plot polygons areas

mergedArea <- c()

areaNameCorrected <- c(

"East Madagascar",

"West Madagascar",

"West Africa",

"Central Africa",

"East/South Africa",

"Central America",

"Northern South America",

"Southern South America",

"West Asia",

"Central/East Asia",

"South Asia",

"Asian islands")

for(i in 1:length(areaName)){

toFortify <- get(paste("area", i, sep="\_"))

toFortify <- fortify(toFortify)

toFortify$group <-paste(areaName[i], toFortify$piece)

toFortify$colour <- gsub("\_", " ", areaName[i],)#areaNameCorrected[i]#colourArea[i]

mergedArea <- rbind(mergedArea, toFortify)

#add to ggplot

#map <- map + geom\_polygon(data = toFortify, aes(x = long, y = lat, group = group), col = colourArea[i], fill = colourArea[i])

}

centroid.sf <- st\_as\_sf(centroid, coords=c("long", "lat"), crs=st\_crs(world))

library(sfheaders)

mergedArea.sf <- sfheaders::sf\_polygon( obj = mergedArea

, x = "long"

, y = "lat"

, polygon\_id = "group"

)

sf::st\_crs(mergedArea.sf) <- st\_crs(world)

#Readd colour

mergedArea.sf$colour <- unique(mergedArea[,c(7,8)])[,2]

# mergedArea, coords=c("long", "lat"), crs=st\_crs(world))

# st\_cast(mergedArea.sf, to="POLYGON")

map <- map +

geom\_sf(data = mergedArea.sf, aes(fill = colour)) +

geom\_sf\_label(data = centroid.sf, aes(label = 1:nrow(centroid)),

fill = c(colourArea[11], colourArea[12], rep("white", times=10)),

col = c("black", "white", rep("black", times=10)),#colourArea[3:12]),

nudge\_y=centroid$nudge\_y,

label.r = unit(0.4, "lines"),

size = 2) +

coord\_sf(crs = proj.map, expand = F) +

scale\_fill\_manual(values = colourArea) +

labs(fill = "") +

theme(

text = element\_text(size = 10),

legend.key = element\_rect(size = 4),

legend.key.size = unit(1, 'lines'),

legend.position="bottom")

map

# # #Just to see the difference as pinpointed by the warning

# # # geosphere::areaPolygon(areaTransitory)

# # # gArea(areaTransitory)

# # # really false

# # # geosphere::centroid(areaTransitory)

# # # gCentroid(areaTransitory)

# # #ok

# #

# # layout(mat=t(c(1,2)), widths=c(40,40), heights=c(40))

# # par(mar=c(0, 0, 0, 0), mgp=c(2, 0.5, 0), xpd=TRUE)

# #

# # Add figure interaction here if needed

# ####

# ## Fig map

# ####

#

# #Create the map of the geographic area

# #Have background

# maps::map("world", fill=TRUE, col="lightgray", bg="white", border=NA, mar = c(0, 0, 0, 0))#, ylim=c(-60, 50))

#

# # addLabel(x=0.05, y=0.075, label="A", radius=7, circle=TRUE, circle.bg="black", font.col="white")

# #

#

# for(i in 1:length(areaName)){

# plot(get(paste("area", i, sep="\_")), col=colourArea[i], border=colourArea[i], add=TRUE) #border="black",

# }

# #Have borders

# #plot(worldMap, col=NA, border="white",bg="white", lwd=0.1, add=TRUE)

# # for(i in 1:length(areaName)){

# # plot(get(paste("area", i, sep="\_")), col=NA, border="black", add=TRUE)

# # }

# points(x=centroid[,1], y=centroid[,2], pch=19, col=c(colourArea[1], colourArea[2], rep("white", times=10)), cex=1.3)

# points(x=centroid[,1], y=centroid[,2], cex=1.3, col=c("white", "white", rep("black", times=10)))

# text(x=centroid[,1], y=centroid[,2], labels=1:length(areaName), cex=0.5, col=c("white", "white", rep("black", times=10)), adj=c(0.5,0.5))

```

\newpage

```{r phylogeny, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=7, fig.height=7, fig.cap=paste("Current frugivorous primate co-occurrence pattern and phylogeny | Primate phylogeny from a consensus tree of 1000 possible trees from the 10kTrees project is depicted in the centre, together with abbreviated species name. The corresponding non-abbreviated names can be found using Appendix Figure \\@ref(fig:figdata). Co-occurring frugivorous (based on a frugivory threshold of ", frugivoryThresholdVector[1], "% and folivory of ", folivoryThresholdVector[1], "%) species are linked by lightgray lines. The geographic area occupied by a species is depicted by the coloured rectangles. Presence was assed given an overlap between the species range and the geographic area of ", geographicThresholdVector[1]\*100, "%.", sep="")}

# ####

# ## Fig interaction

# ####

#Tree

tree <- read.tree("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Raw\_data/Tree/Tree\_diet.nex")

tree <- drop.tip(tree,

tree $tip.label[

which(tree$tip.label

%nin% summaryDataForPlot$SpeciesForPhylogeny)])

#Create geography df

geoBinary <- as.data.frame(summaryDataForPlot[,c(3,which(colnames(summaryDataForPlot)=="geographicCode"))])

colnames(geoBinary) <- c("SpeciesPhylo", "Loc")

#Create species ID

hc = as.hclust(tree)#bird.orders)

labels = hc$labels # name of birds

labels.tordc <- as.data.frame(labels)

colnames(labels.tordc) <- "Name"

labels.tordc <- separate(labels.tordc, col="Name", into=c("Name1", "Name2", "Name3"), sep="\_")

labels.rdc <- apply(labels.tordc, 1, function(x){

if(!is.na(x[3])){

paste(toupper(substr(x[1], 1, 1)), ". ", substr(x[2], 1, 3), ". ", substr(x[3], 1, 1), ".", sep="")

} else{paste(toupper(substr(x[1], 1, 1)), ". ", substr(x[2], 1, 3), sep="")

}

}

)

#Match to have right order for geography

locationSpecies <- geoBinary$Loc[match(labels, geoBinary$SpeciesPhylo)]

colLoc <- colourArea

#Match to have diet

dietSpecies <- summaryDataForPlot$DietaryGuild[match(labels, summaryDataForPlot$SpeciesForPhylogeny)]

#Getting species labels abbreviated

speciesLabels <- hc$labels#Should be in the tree order

#Create the circos plot linking species based on their diet and geography

circos.clear()

circos.par(gap.degree=0, gap.after=0, cell.padding=c(0,0,0,0), track.margin = c(0, 0), "canvas.xlim" = c(-1.1, 1.1), "canvas.ylim" = c(-1.1, 1.1))

circos.initialize(speciesLabels, xlim = c(0, 1))

# #Add family

# circos.track(ylim = c(0, 1), bg.border = NA, track.height = 0.1, track.margin=c(0.01, 0.01),

# panel.fun = function(x, y) {

# i=CELL\_META$sector.numeric.index

# circos.rect(CELL\_META$cell.xlim[1],CELL\_META$cell.ylim[1], CELL\_META$cell.xlim[2],CELL\_META$cell.ylim[2], col="white", border=NA)

# })

#

# circos.track(ylim = c(0, 1), bg.border = NA, track.height = 0.1, track.margin=c(0.01, 0.01),

# panel.fun = function(x, y) {

# i=CELL\_META$sector.numeric.index

# circos.rect(CELL\_META$cell.xlim[1],CELL\_META$cell.ylim[1], CELL\_META$cell.xlim[2],CELL\_META$cell.ylim[2], col="white", border=NA)

# })

#

# for(i in 1:length(unique(summaryDataForPlot$Family))){

# speciesForPlotFamily <- speciesLabels[speciesLabels %in% summaryDataForPlot$SpeciesForPhylogeny[summaryDataForPlot$Family %in% unique(summaryDataForPlot$Family)[i]]]

# if(length(speciesForPlotFamily) > 0){

# highlight.sector(speciesForPlotFamily, track.index = 1, col = adjustcolor("black", alpha.f=0.15), padding = c(0, 0.2, 0, 0),

# text = unique(summaryDataForPlot$Family)[i], cex = 0.5, text.col = "white", niceFacing = TRUE)

# }

# }

#Species name + area it belongs to

circos.track(ylim = c(0, 1), bg.border = NA, track.height = 0.1, track.margin=c(0.01, 0.01),

panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

circos.text(CELL\_META$xcenter, 1, labels.rdc[i], adj = c(0, 0),

facing = "clockwise", niceFacing = TRUE,

col = "black", cex = 0.75, font=3)

geo <- as.numcharac(unlist(strsplit(locationSpecies[i], "")))

for(g in 1:length(geo)){

if(geo[g]==1){

#circos.points(CELL\_META$xcenter,0.75/length(geo)\*(2\*g-1)/2, col=as.character(colLoc[g]), pch=19, cex=0.2)

circos.rect(CELL\_META$cell.xlim[1],0.75/length(geo)\*(g-1), CELL\_META$cell.xlim[2], 0.75/length(geo)\*(g), col=as.character(colLoc[g]), border=NA)

}

}

})

#Plot the geographic links

for(i in 1:length(speciesLabels)){

#locI <- which(strsplit(locationSpecies[i], "")==1)

for(j in i:length(speciesLabels)){

#locJ <- which(strsplit(locationSpecies[j], "")==1)

product <- as.numcharac(unlist(strsplit(locationSpecies[j], "")))\*as.numcharac(unlist(strsplit(locationSpecies[i], "")))

if(i==j|(length(unique(product))==1&unique(product)[1]==0)){

#Do nothing

}

else{

if(dietSpecies[i]=="Fruit"&dietSpecies[i]==dietSpecies[j]){

# colour <- as.data.frame(table(colLoc[which(product==1)]))

# if(is.finite(max(colour$Freq))){

# }else{

# print(c(i,j))

# }

# colour <- colour[colour$Freq==max(colour$Freq),1][1]

circos.link(speciesLabels[i], runif(1, 0, 1), speciesLabels[j], runif(1, 0, 1), lwd=0.2, col=adjustcolor("black",alpha.f=0.15))#"lightgray")#adjustcolor(as.character(colour), alpha.f=0.9))

}

else{

#circos.link(speciesLabels[i], runif(1, 0, 1), speciesLabels[j], runif(1, 0, 1), lwd=1, col="lightgray")

}

}

}

}

circos.clear()

#Plot the phylogenetic tree in a new circular plot

n = length(labels) # number of species

dend = as.dendrogram(hc)

par(new = TRUE) # <- magic

circos.par("canvas.xlim" = c(-1.05, 1.05), "canvas.ylim" = c(-1.25, 1.25))

circos.initialize("a", xlim = c(0, n)) # only one sector

# circos.track(ylim = c(0, 1), bg.border = NA, track.height = 0.3,

# panel.fun = function(x, y) {

# for(i in seq\_len(n)) {

# circos.text(i-0.5, 0, labels.rdc[i], adj = c(0, 0.5),

# facing = "clockwise", niceFacing = TRUE,

# col = "black", cex = 0.2, font=3)

# }

# })

#suppressPackageStartupMessages(library(dendextend))

#dend = color\_branches(dend, k = 6, col = 1:6)

dend\_height = attr(dend, "height")

circos.track(ylim = c(0, dend\_height), bg.border = NA,

track.height = 0.95, panel.fun = function(x, y) {

circos.dendrogram(dend)

})

circos.clear()

```

\newpage

```{r figbrain, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=7, fig.height=7, fig.cap=paste("(Left) EQ or relative brain size value among frugivorous primates (Right) Studied brain areas | (Left) The circular rows are indicated by the colours which match a specific brain area. The darker background emphasises when values are above average, while the lighter background emphasises when values are below average. The mean value (after scaling and based on one random sampling among possible values, but see \\@ref(fig:figvariabilitydata) for visualization of measure variability) for the Encephalization Quotient (EQ) or relative size of brain parts, when available, is depicted by a plain circle for frugivorous species. The frugivorous threshold was fixed to ", frugivoryThresholdVector[1], "% and folivory to ", folivoryThresholdVector[1], "%. (Right) A 3D brain from \*Homo sapiens\* is depicted (\*neurobase\* package [@neurobase], \*misc3d\* package [@misc3d]). The arrows indicate the sagital and frontal axes. Studied brain area are coloured, although the neocortex was not coloured for readability since it corresponds to the external layer of the cerebral hemisphere. In short, the MOB is involved in immediate olfactory information processing, the Neocortex and the Cerebellum support a working memory and memory consolidation processes [@wiltgen2004new; @koziol2014consensus; @sokolov2017cerebellum], and the Hippocampus supports a working memory and a long-term spatio-temporal memory [@burgess2002human]. The Striatum is involved in social information processing [@izuma2008processing].", sep="")}

library(RColorBrewer)

colourVector <- c("darkgrey", brewer.pal(n = 5, name = "Set1")[1:5])

colourVectorbis <- c("lightgray", brewer.pal(n = 5, name = "Pastel1")[1:5])

colour.circle.points <- c("black", "darkred", "darkblue", "darkgreen", "purple4", "orange4")

###

## Fig brain values / circular

###

summaryDataForPlot$EQ <- summaryDataForPlot$Brain\*1.036\*(10\*\*-3)/(0.085\*summaryDataForPlot$Bodymass\*\*0.775)

summaryDataForPlot$ratioNeocortex <- summaryDataForPlot$Neocortex/summaryDataForPlot$Bodymass

summaryDataForPlot$ratioHippocampus <- summaryDataForPlot$Hippocampus/summaryDataForPlot$Bodymass

summaryDataForPlot$ratioCerebellum <- summaryDataForPlot$Cerebellum/summaryDataForPlot$Bodymass

summaryDataForPlot$ratioStriatum <- summaryDataForPlot$Striatum/summaryDataForPlot$Bodymass

summaryDataForPlot$ratioMOB <- summaryDataForPlot$MOB/summaryDataForPlot$Bodymass

#Brain data

relativeValueEQ <- scale(summaryDataForPlot$EQ[match(speciesLabels, summaryDataForPlot$SpeciesForPhylogeny)])#runif(length(speciesLabels), -1, 1)

relativeValueNeocortex <- scale(summaryDataForPlot$ratioNeocortex[match(speciesLabels, summaryDataForPlot$SpeciesForPhylogeny)])

relativeValueHippocampus <- scale(summaryDataForPlot$ratioHippocampus[match(speciesLabels, summaryDataForPlot$SpeciesForPhylogeny)])

relativeValueCerebellum <- scale(summaryDataForPlot$ratioCerebellum[match(speciesLabels, summaryDataForPlot$SpeciesForPhylogeny)])

relativeValueStriatum <- scale(summaryDataForPlot$ratioStriatum[match(speciesLabels, summaryDataForPlot$SpeciesForPhylogeny)])

relativeValueMOB <- scale(summaryDataForPlot$ratioMOB[match(speciesLabels, summaryDataForPlot$SpeciesForPhylogeny)])

layout(mat=cbind(c(1,1), c(2,3)), widths=c(35,15), heights=c(15,35))

par(mar=c(0, 0, 0, 0), mgp=c(2, 0.5, 0), xpd=TRUE)

# plot(0, 0, type="n")

library(circlize)

circos.clear()

circos.par(gap.degree=0, gap.after=0, cell.padding=c(0,0,0,0), track.margin=c(0, 0))

circos.initialize(speciesLabels, xlim = c(0, 1))

#Species name

circos.track(ylim = c(0, 20), bg.border = NA, track.height = 0.05, track.margin=c(0.01, 0.1),

panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

circos.text(CELL\_META$xcenter, 0, labels.rdc[i], adj = c(0, 0),

facing = "clockwise", niceFacing = TRUE,

col = "black", cex = 0.6, font=3)

})

#

#Background

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVector[1], border=colourVector[1])

}, track.height = 1/15)

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVectorbis[1], border=colourVectorbis[1])

}, track.height = 1/15)

#Background

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVector[2], border=colourVector[2])

}, track.height = 1/15)

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVectorbis[2], border=colourVectorbis[2])

}, track.height = 1/15)

#Background

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVector[3], border=colourVector[3])

}, track.height = 1/15)

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVectorbis[3], border=colourVectorbis[3])

}, track.height = 1/15)

#Background

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVector[4], border=colourVector[4])

}, track.height = 1/15)

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVectorbis[4], border=colourVectorbis[4])

}, track.height = 1/15)

#Background

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVector[5], border=colourVector[5])

}, track.height = 1/15)

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVectorbis[5], border=colourVectorbis[5])

}, track.height = 1/15)

#Background

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVector[6], border=colourVector[6])

}, track.height = 1/15)

circos.track(ylim = c(0, 1), bg.border = NA, panel.fun = function(x, y) {

circos.rect(0, 0, 1, 1, col=colourVectorbis[6], border=colourVectorbis[6])

}, track.height = 1/15)

library(plotrix)

#Main circle

for(i in 1:13){

draw.circle(x=0,y=0,0.91-1/15-(i-1)\*1/15, col=NA, border="white")

}

#increment of 0.5

for(i in 1:26){

draw.circle(x=0,y=0,0.91-1/15-(i-1)\*1/15/2, col=NA, border="white", lty=2)

}

#Value

#EQ

absMax <- max(abs(relativeValueEQ), na.rm=TRUE)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=2, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

#circos.rect(0, 0, 1, 1, col=colourPositive, border=colourPositive)

if(is.na(relativeValueEQ[i])){} else{

if(relativeValueEQ[i] > 0 & dietSpecies[i]=="Fruit"){

circos.points(CELL\_META$xcenter, relativeValueEQ[i]/absMax, pch=19, col=colour.circle.points[1], cex=0.7)

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueEQ[i]/absMax, col=colour.circle.points[1], lty=3)

}

else if(relativeValueEQ[i] > 0 & dietSpecies[i]=="Leaf"){

circos.points(CELL\_META$xcenter, relativeValueEQ[i]/absMax, pch=21, col=colour.circle.points[1], bg="white", cex=0.7)

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueEQ[i]/absMax, col=colour.circle.points[1], lty=3)

}

else{}

}

}, track.height = 0.1)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=3, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

if(is.na(relativeValueEQ[i])){} else{

#circos.rect(0, 0, 1, 1, col=colourNegative, border=colourNegative)

if(relativeValueEQ[i] <= 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueEQ[i]/absMax, col=colour.circle.points[1], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueEQ[i]/absMax, pch=19, col=colour.circle.points[1], cex=0.7)

}

else if(relativeValueEQ[i] <= 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueEQ[i]/absMax, col=colour.circle.points[1], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueEQ[i]/absMax, pch=21, col=colour.circle.points[1], bg="white", cex=0.7)

}

else{}

}

}, track.height = 0.1)

#Striatum

absMax <- max(abs(relativeValueStriatum), na.rm=TRUE)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=4, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

#circos.rect(0, 0, 1, 1, col=colourPositive, border=colourPositive)

if(is.na(relativeValueStriatum[i])){} else{

if(relativeValueStriatum[i] > 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueStriatum[i]/absMax, col=colour.circle.points[2], lty=3)

circos.points(CELL\_META$xcenter, relativeValueStriatum[i]/absMax, pch=19, col=colour.circle.points[2], cex=0.65)

}

else if(relativeValueStriatum[i] > 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueStriatum[i]/absMax, col=colour.circle.points[2], lty=3)

circos.points(CELL\_META$xcenter, relativeValueStriatum[i]/absMax, pch=21, col=colour.circle.points[2], bg="white", cex=0.65)

}

else{}

}

}, track.height = 0.1)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=5, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

if(is.na(relativeValueStriatum[i])){} else{

#circos.rect(0, 0, 1, 1, col=colourNegative, border=colourNegative)

if(relativeValueStriatum[i] <= 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueStriatum[i]/absMax, col=colour.circle.points[2], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueStriatum[i]/absMax, pch=19, col=colour.circle.points[2], cex=0.65)

}

else if(relativeValueStriatum[i] <= 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueStriatum[i]/absMax, col=colour.circle.points[2], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueStriatum[i]/absMax, pch=21, col=colour.circle.points[2], bg="white", cex=0.65)

}

else{}

}

}, track.height = 0.1)

#MOB

absMax <- max(abs(relativeValueMOB), na.rm=TRUE)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=6, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

#circos.rect(0, 0, 1, 1, col=colourPositive, border=colourPositive)

if(is.na(relativeValueMOB[i])){} else{

if(relativeValueMOB[i] > 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueMOB[i]/absMax, col=colour.circle.points[3], lty=3)

circos.points(CELL\_META$xcenter, relativeValueMOB[i]/absMax, pch=19, col=colour.circle.points[3], cex=0.65)

}

else if(relativeValueMOB[i] > 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueMOB[i]/absMax, col=colour.circle.points[3], lty=3)

circos.points(CELL\_META$xcenter, relativeValueMOB[i]/absMax, pch=21, col=colour.circle.points[3], bg="white", cex=0.65)

}

else{}

}

}, track.height = 0.1)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=7, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

if(is.na(relativeValueMOB[i])){} else{

#circos.rect(0, 0, 1, 1, col=colourNegative, border=colourNegative)

if(relativeValueMOB[i] <= 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueMOB[i]/absMax, col=colour.circle.points[3], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueMOB[i]/absMax, pch=19, col=colour.circle.points[3], cex=0.65)

}

else if(relativeValueMOB[i] <= 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueMOB[i]/absMax, col=colour.circle.points[3], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueMOB[i]/absMax, pch=21, col=colour.circle.points[3], bg="white", cex=0.65)

}

else{}

}

}, track.height = 0.1)

#Neocortex

absMax <- max(abs(relativeValueNeocortex), na.rm=TRUE)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=8, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

#circos.rect(0, 0, 1, 1, col=colourPositive, border=colourPositive)

if(is.na(relativeValueNeocortex[i])){} else{

if(relativeValueNeocortex[i] > 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueNeocortex[i]/absMax, col=colour.circle.points[4], lty=3)

circos.points(CELL\_META$xcenter, relativeValueNeocortex[i]/absMax, pch=19, col=colour.circle.points[4], cex=0.6)

}

else if(relativeValueNeocortex[i] > 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueNeocortex[i]/absMax, col=colour.circle.points[4], lty=3)

circos.points(CELL\_META$xcenter, relativeValueNeocortex[i]/absMax, pch=21, col=colour.circle.points[4], bg="white", cex=0.6)

}

else{}

}

}, track.height = 0.1)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=9, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

if(is.na(relativeValueNeocortex[i])){} else{

#circos.rect(0, 0, 1, 1, col=colourNegative, border=colourNegative)

if(relativeValueNeocortex[i] <= 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueNeocortex[i]/absMax, col=colour.circle.points[4], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueNeocortex[i]/absMax, pch=19, col=colour.circle.points[4], cex=0.6)

}

else if(relativeValueNeocortex[i] <= 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueNeocortex[i]/absMax, col=colour.circle.points[4], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueNeocortex[i]/absMax, pch=21, col=colour.circle.points[4], bg="white", cex=0.6)

}

else{}

}

}, track.height = 0.1)

#Hippocampus

absMax <- max(abs(relativeValueHippocampus), na.rm=TRUE)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=10, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

#circos.rect(0, 0, 1, 1, col=colourPositive, border=colourPositive)

if(is.na(relativeValueHippocampus[i])){} else{

if(relativeValueHippocampus[i] > 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueHippocampus[i]/absMax, col=colour.circle.points[5], lty=3)

circos.points(CELL\_META$xcenter, relativeValueHippocampus[i]/absMax, pch=19, col=colour.circle.points[5], cex=0.55)

}

else if(relativeValueHippocampus[i] > 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueHippocampus[i]/absMax, col=colour.circle.points[5], lty=3)

circos.points(CELL\_META$xcenter, relativeValueHippocampus[i]/absMax, pch=21, col=colour.circle.points[5], bg="white", cex=0.55)

}

else{}

}

}, track.height = 0.1)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=11, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

if(is.na(relativeValueHippocampus[i])){} else{

#circos.rect(0, 0, 1, 1, col=colourNegative, border=colourNegative)

if(relativeValueHippocampus[i] <= 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueHippocampus[i]/absMax, col=colour.circle.points[5], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueHippocampus[i]/absMax, pch=19, col=colour.circle.points[5], cex=0.55)

}

else if(relativeValueHippocampus[i] <= 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueHippocampus[i]/absMax, col=colour.circle.points[5], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueHippocampus[i]/absMax, pch=21, col=colour.circle.points[5], bg="white", cex=0.55)

}

else{}

}

}, track.height = 0.1)

#Cerebellum

absMax <- max(abs(relativeValueCerebellum), na.rm=TRUE)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=12, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

#circos.rect(0, 0, 1, 1, col=colourPositive, border=colourPositive)

if(is.na(relativeValueCerebellum[i])){} else{

if(relativeValueCerebellum[i] > 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueCerebellum[i]/absMax, col=colour.circle.points[6], lty=3)

circos.points(CELL\_META$xcenter, relativeValueCerebellum[i]/absMax, pch=19, col=colour.circle.points[6], cex=0.5)

}

else if(relativeValueCerebellum[i] > 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 0, CELL\_META$xcenter, relativeValueCerebellum[i]/absMax, col=colour.circle.points[6], lty=3)

circos.points(CELL\_META$xcenter, relativeValueCerebellum[i]/absMax, pch=21, col=colour.circle.points[6], bg="white", cex=0.5)

}

else{}

}

}, track.height = 0.1)

circos.track(ylim = c(0, 1), bg.border = NA, track.index=13, panel.fun = function(x, y) {

i=CELL\_META$sector.numeric.index

if(is.na(relativeValueCerebellum[i])){} else{

#circos.rect(0, 0, 1, 1, col=colourNegative, border=colourNegative)

if(relativeValueCerebellum[i] <= 0 & dietSpecies[i]=="Fruit"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueCerebellum[i]/absMax, col=colour.circle.points[6], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueCerebellum[i]/absMax, pch=19, col=colour.circle.points[6], cex=0.5)

}

else if(relativeValueCerebellum[i] <= 0 & dietSpecies[i]=="Leaf"){

circos.segments(CELL\_META$xcenter, 1, CELL\_META$xcenter, 1 + relativeValueCerebellum[i]/absMax, col=colour.circle.points[6], lty=3)

circos.points(CELL\_META$xcenter, 1 + relativeValueCerebellum[i]/absMax, pch=21, col=colour.circle.points[6], bg="white", cex=0.5)

}

else{}

}

}, track.height = 0.1)

#Empty plot

par(mar=c(0, 0, 0, 0), mgp=c(2, 0.5, 0), xpd=TRUE)

plot(0, 0, xlim=c(0,1), ylim=c(0,1), xlab="", ylab="", las=1, type="n", tcl=-0.25, bty="n",

xaxt="n",xaxs="i",yaxs="i", yaxt="n", xpd=TRUE)

# #Add brain

library(png)

brainIMG <- readPNG("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Plots/3dplot.png")

addImg(brainIMG, x = 0.425, y = 0.65, width = 1.3)

#Add circle contour

draw.circle(x=0.45,y=0.55,0.435, col=NA, border="lightgray", lwd=2)

draw.circle(x=0.45,y=0.55,0.405, col=NA, border="lightgray", lwd=2)

par(mar=c(0, 0, 0, 0), mgp=c(2, 0.5, 0), xpd=TRUE)

plot(0, 0, xlim=c(0,1), ylim=c(0,1), xlab="", ylab="", las=1, type="n", tcl=-0.25, bty="n",

xaxt="n",xaxs="i",yaxs="i", yaxt="n", xpd=TRUE)

colourHip=colourVector[5]

colourCereb=colourVector[6]

colourOlf=colourVector[3]

colourStri=colourVector[2]

legend(x=0.1, y=0.8, legend = c("EQ", "Striatum", "MOB", "Hippocampus", "Neocortex", "Cerebellum", "Frugivorous species", "Folivorous species"), cex = 1, fill = c(colourVector, NA, NA), pch=c(NA, NA, NA, NA, NA, NA, 19, 1), border=c("black", "black", "black", "black", "black", "black", "white", "white"), col=c(NA, NA, NA, NA, NA, NA, "black", "black"), bty="n", seg.len=0.75)

```

\newpage

```{r prepfigevolution}

###Set working directory

setwd("C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates")

summaryBrainFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

summaryEQFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

summaryNeocortexFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

summaryHippocampusFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

summaryCerebellumFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

summaryStriatumFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

summaryMOBFrugivory <- as.data.frame(matrix(NA, nrow=10\*(repetition+1), ncol=53))

counter=0

start=counter

end=counter

numberSimulations=10

for(a in 1:2){

for(b in 1:2){

for(c in 1:2){

for(d in 1:10){

counter=end+1

start=counter

end=counter + numberSimulations - 1

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_BrainBodymassRaw",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryBrainFrugivory[start:end,] <- as.data.frame(toAdd)

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_EQ",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryEQFrugivory[start:end,] <- toAdd

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_NeocortexBodymassRaw",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryNeocortexFrugivory[start:end,] <- toAdd

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_HippocampusBodymassRaw",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryHippocampusFrugivory[start:end,] <- toAdd

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_CerebellumBodymassRaw",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryCerebellumFrugivory[start:end,] <- toAdd

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_StriatumBodymassRaw",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryStriatumFrugivory[start:end,] <- toAdd

}, error=function(e){

#Do nothing

}

)

tryCatch(

{toAdd <- read.delim(paste("Processed\_data/OutputEvolModel/Output\_evolutionary\_history\_MOBBodymassRaw",a,"\_",b,"\_",c,"\_",d,".txt", sep=""))

summaryMOBFrugivory[start:end,] <- toAdd

}, error=function(e){

#Do nothing

}

)

}

}

}

}

summaryBrainFrugivory <- summaryBrainFrugivory[!is.na(summaryBrainFrugivory[,1]),]

summaryEQFrugivory <- summaryEQFrugivory[!is.na(summaryEQFrugivory[,1]),]

summaryNeocortexFrugivory <- summaryNeocortexFrugivory[!is.na(summaryNeocortexFrugivory[,1]),]

summaryHippocampusFrugivory <- summaryHippocampusFrugivory[!is.na(summaryHippocampusFrugivory[,1]),]

summaryCerebellumFrugivory <- summaryCerebellumFrugivory[!is.na(summaryCerebellumFrugivory[,1]),]

summaryStriatumFrugivory <- summaryStriatumFrugivory[!is.na(summaryStriatumFrugivory[,1]),]

summaryMOBFrugivory <- summaryMOBFrugivory[!is.na(summaryMOBFrugivory[,1]),]

colnames(summaryBrainFrugivory) <- colnames(toAdd)

colnames(summaryEQFrugivory) <- colnames(toAdd)

colnames(summaryNeocortexFrugivory) <- colnames(toAdd)

colnames(summaryHippocampusFrugivory) <- colnames(toAdd)

colnames(summaryCerebellumFrugivory) <- colnames(toAdd)

colnames(summaryStriatumFrugivory) <- colnames(toAdd)

colnames(summaryMOBFrugivory) <- colnames(toAdd)

##----

colNum <-c("darkgrey", brewer.pal(n = 5, name = "Set1")[1:5])

models <- c("BM", "OU", "EB", "MC", expression(DD[italic(lin)]), expression(DD[italic(exp)]))

colourModels <- brewer.pal(n = 6, name = "Set1")

```

```{r figresultsevolution, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=7, fig.height=6, fig.cap="AICc weights of fitted models of trait evolution for each brain part | Plotted is the AICc weight, a measure of relative support for a given model, for non-competitive (BM, OU, EB) and competitive (MC, DD$\_{lin}$, DD$\_{exp}$) models. The points represent the average AICc weight obtained (when considering the six models from a same run), while the vertical bars indicate the standard deviation given all tested conditions (see [Models of trait evolution: does interspecific interactions shape brain size evolution?])."}

layout(mat=rbind(c(1,2,3), c(4,5,6)), widths=c(5,5,5), heights=c(5,5))

par(mar=c(3.5, 3, 2, 0.5), mgp=c(2, 0.5, 0), xpd=TRUE)

#note: 1= second run for frugivory 20%

#note: \_2= first run for frugivory 20%

## EQ

plot(

x=0, y=0, xlab="", ylab="AICc weight", cex.sub=1.6,

xlim=c(0,7), ylim=c(0,1.2),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=7, xintsmall=0.25, xintbig=1, ymin=0, ymax=1, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=2, tcl=-0.25)

text(x=1:6, y=rep(-0.1, times=6), labels=models, cex=0.8, xpd=TRUE)

for(i in 1:6){

meanPt <- mean(as.numcharac(summaryEQFrugivory[, ncol(summaryEQFrugivory)-6+i]))

sd <- sd(as.numcharac(summaryEQFrugivory[, ncol(summaryEQFrugivory)-6+i]))

#sd <- sd/nrow(summaryEQFrugivory) #error not sd

errorBars(location=i, meanPt=meanPt, barValue=sd, refUnit=1, col="black", minValue=0, maxValue=1, horiz=FALSE)

points(x=i, y=meanPt, pch=19, col=colourModels[i], xpd=TRUE)

}

#b and r are the rate for density dependance (DD) of the speciation rate. If positive, positive DD, otherwise, negative.

#add their values below:

text(x=c(4,5,6), y=c(-0.2, -0.2),

labels=c(

"r ~",

round(mean(as.numcharac(summaryEQFrugivory$DDlingeo.b)), digit=3),

round(mean(as.numcharac(summaryEQFrugivory$DDexpgeo.r)), digit=3)

), cex=0.75, xpd=TRUE)

draw.circle(x=0.3,y=1.1,0.25, col=colNum[1], border=NA)

text(x=0.3, y=1.1, labels="1", xpd=TRUE, col="white", font=2)

text(x=3.5, y=1.1, labels="EQ", xpd=TRUE, col="black", font=2, cex=1.5)

##-------------

##------------

#Striatum

plot(

x=0, y=0, xlab="", ylab="", cex.sub=1.6,

xlim=c(0,7), ylim=c(0,1.2),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=7, xintsmall=0.25, xintbig=1, ymin=0, ymax=1, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

#axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=2, tcl=-0.25)

segments(x0 = -1, x1 = -1, y0 = 0, y1 = 1, lty = 2, col = colNum[2])

text(x=1:6, y=rep(-0.1, times=6), labels=models, cex=0.8, xpd=TRUE)

for(i in 1:6){

meanPt <- mean(as.numcharac(summaryStriatumFrugivory[, ncol(summaryStriatumFrugivory)-6+i]))

sd <- sd(as.numcharac(summaryStriatumFrugivory[, ncol(summaryStriatumFrugivory)-6+i]))

#sd <- sd/nrow(summaryStriatumFrugivory) #error not sd

errorBars(location=i, meanPt=meanPt, barValue=sd, refUnit=1, col="black", minValue=0, maxValue=1, horiz=FALSE)

points(x=i, y=meanPt, pch=19, col=colourModels[i], xpd=TRUE)

}

draw.circle(x=0.3,y=1.1,0.25, col=colNum[2], border=NA)

text(x=0.3, y=1.1, labels="2", xpd=TRUE, col="white", font=2)

text(x=3.5, y=1.1, labels="Striatum", xpd=TRUE, col="black", font=2, cex=1.5)

#b and r are the rate for density dependance (DD) of the speciation rate. If positive, positive DD, otherwise, negative.

#add their values below:

text(x=c(4,5,6), y=c(-0.2, -0.2),

labels=c(

"r ~",

round(mean(as.numcharac(summaryStriatumFrugivory$DDlingeo.b)), digit=3),

round(mean(as.numcharac(summaryStriatumFrugivory$DDexpgeo.r)), digit=3)

), cex=0.75, xpd=TRUE)

##------------

##-------------

#MOB

plot(

x=0, y=0, xlab="", ylab="", cex.sub=1.6,

xlim=c(0,7), ylim=c(0,1.2),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=7, xintsmall=0.25, xintbig=1, ymin=0, ymax=1, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

#axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=2, tcl=-0.25)

segments(x0 = -1, x1 = -1, y0 = 0, y1 = 1, lty = 2, col = colNum[3])

text(x=1:6, y=rep(-0.1, times=6), labels=models, cex=0.8, xpd=TRUE)

for(i in 1:6){

meanPt <- mean(as.numcharac(summaryMOBFrugivory[, ncol(summaryMOBFrugivory)-6+i]))

sd <- sd(as.numcharac(summaryMOBFrugivory[, ncol(summaryMOBFrugivory)-6+i]))

#sd <- sd/nrow(summaryMOBFrugivory) #error not sd

errorBars(location=i, meanPt=meanPt, barValue=sd, refUnit=1, col="black", minValue=0, maxValue=1, horiz=FALSE)

points(x=i, y=meanPt, pch=19, col=colourModels[i], xpd=TRUE)

}

draw.circle(x=0.3,y=1.1,0.25, col=colNum[3], border=NA)

text(x=0.3, y=1.1, labels="3", xpd=TRUE, col="white", font=2)

text(x=3.5, y=1.1, labels="MOB", xpd=TRUE, col="black", font=2, cex=1.5)

#b and r are the rate for density dependance (DD) of the speciation rate. If positive, positive DD, otherwise, negative.

#add their values below:

text(x=c(4,5,6), y=c(-0.2, -0.2),

labels=c(

"r ~",

round(mean(as.numcharac(summaryMOBFrugivory$DDlingeo.b)), digit=3),

round(mean(as.numcharac(summaryMOBFrugivory$DDexpgeo.r)), digit=3)

), cex=0.75, xpd=TRUE)

##------------

##------------

#Hippocampus

plot(

x=0, y=0, xlab="", ylab="AICc weight", cex.sub=1.6,

xlim=c(0,7), ylim=c(0,1.2),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=7, xintsmall=0.25, xintbig=1, ymin=0, ymax=1, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=2, tcl=-0.25)

text(x=1:6, y=rep(-0.1, times=6), labels=models, cex=0.8, xpd=TRUE)

for(i in 1:6){

meanPt <- mean(as.numcharac(summaryHippocampusFrugivory[, ncol(summaryHippocampusFrugivory)-6+i]))

sd <- sd(as.numcharac(summaryHippocampusFrugivory[, ncol(summaryHippocampusFrugivory)-6+i]))

#sd <- sd/nrow(summaryHippocampusFrugivory) #error not sd

errorBars(location=i, meanPt=meanPt, barValue=sd, refUnit=1, col="black", minValue=0, maxValue=1, horiz=FALSE)

points(x=i, y=meanPt, pch=19, col=colourModels[i], xpd=TRUE)

}

draw.circle(x=0.3,y=1.1,0.25, col=colNum[4], border=NA)

text(x=0.3, y=1.1, labels="6", xpd=TRUE, col="white", font=2)

text(x=3.5, y=1.1, labels="Hippocampus", xpd=TRUE, col="black", font=2, cex=1.5)

#b and r are the rate for density dependance (DD) of the speciation rate. If positive, positive DD, otherwise, negative.

#add their values below:

text(x=c(4,5,6), y=c(-0.2, -0.2),

labels=c(

"r ~",

round(mean(as.numcharac(summaryHippocampusFrugivory$DDlingeo.b)), digit=3),

round(mean(as.numcharac(summaryHippocampusFrugivory$DDexpgeo.r)), digit=3)

), cex=0.75, xpd=TRUE)

##------------

##-------------

#Neocortex

plot(

x=0, y=0, xlab="", ylab="", cex.sub=1.6,

xlim=c(0,7), ylim=c(0,1.2),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=7, xintsmall=0.25, xintbig=1, ymin=0, ymax=1, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

#axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=2, tcl=-0.25)

segments(x0 = -1, x1 = -1, y0 = 0, y1 = 1, lty = 2, col = colNum[5])

text(x=1:6, y=rep(-0.1, times=6), labels=models, cex=0.8, xpd=TRUE)

for(i in 1:6){

meanPt <- mean(as.numcharac(summaryNeocortexFrugivory[, ncol(summaryNeocortexFrugivory)-6+i]))

sd <- sd(as.numcharac(summaryNeocortexFrugivory[, ncol(summaryNeocortexFrugivory)-6+i]))

#sd <- sd/nrow(summaryNeocortexFrugivory) #error not sd

errorBars(location=i, meanPt=meanPt, barValue=sd, refUnit=1, col="black", minValue=0, maxValue=1, horiz=FALSE)

points(x=i, y=meanPt, pch=19, col=colourModels[i], xpd=TRUE)

}

draw.circle(x=0.3,y=1.1,0.25, col=colNum[5], border=NA)

text(x=0.3, y=1.1, labels="5", xpd=TRUE, col="white", font=2)

text(x=3.5, y=1.1, labels="Neocortex", xpd=TRUE, col="black", font=2, cex=1.5)

#b and r are the rate for density dependance (DD) of the speciation rate. If positive, positive DD, otherwise, negative.

#add their values below:

text(x=c(4,5,6), y=c(-0.2, -0.2),

labels=c(

"r ~",

round(mean(as.numcharac(summaryNeocortexFrugivory$DDlingeo.b)), digit=3),

round(mean(as.numcharac(summaryNeocortexFrugivory$DDexpgeo.r)), digit=3)

), cex=0.75, xpd=TRUE)

##-------------

##-------------

#Cerebellum

plot(

x=0, y=0, xlab="", ylab="", cex.sub=1.6,

xlim=c(0,7), ylim=c(0,1.2),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=7, xintsmall=0.25, xintbig=1, ymin=0, ymax=1, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

#axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=2, tcl=-0.25)

segments(x0 = -1, x1 = -1, y0 = 0, y1 = 1, lty = 2, col = colNum[6])

text(x=1:6, y=rep(-0.1, times=6), labels=models, cex=0.8, xpd=TRUE)

for(i in 1:6){

meanPt <- mean(as.numcharac(summaryCerebellumFrugivory[, ncol(summaryCerebellumFrugivory)-6+i]))

sd <- sd(as.numcharac(summaryCerebellumFrugivory[, ncol(summaryCerebellumFrugivory)-6+i]))

#sd <- sd/nrow(summaryCerebellumFrugivory) #error not sd

errorBars(location=i, meanPt=meanPt, barValue=sd, refUnit=1, col="black", minValue=0, maxValue=1, horiz=FALSE)

points(x=i, y=meanPt, pch=19, col=colourModels[i], xpd=TRUE)

}

draw.circle(x=0.3,y=1.1,0.25, col=colNum[6], border=NA)

text(x=0.3, y=1.1, labels="7", xpd=TRUE, col="white", font=2)

text(x=3.5, y=1.1, labels="Cerebellum", xpd=TRUE, col="black", font=2, cex=1.5)

#b and r are the rate for density dependance (DD) of the speciation rate. If positive, positive DD, otherwise, negative.

#add their values below:

text(x=c(4,5,6), y=c(-0.2, -0.2),

labels=c(

"r ~",

round(mean(as.numcharac(summaryCerebellumFrugivory$DDlingeo.b)), digit=3),

round(mean(as.numcharac(summaryCerebellumFrugivory$DDexpgeo.r)), digit=3)

), cex=0.75, xpd=TRUE)

###----------------------

```

\newpage

<!-- TC:ignore -->

# Literature cited

<div id="refs"></div>

\beginsupplement

\newpage

# Appendix

## Data availability

Availability of trait and biogeography range for the `r length(phylo\_init$tip.label)` primate species represented in the primate phylogeny of the 10kTrees project is depicted in Appendix Figure \@ref(fig:figdata).

## Data variability

We present below the results of the assessments of data variability depending on the considered thresholds (for frugivory, folivory or overlap) and the data set that is used, specifically related to bieography ranges, or anatomical/behavioural traits.

### Sensitivity to variation in biogeography range

```{r figcomparison, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=3.5, fig.height=3.5, fig.cap=paste("Percent of species with differently identified biogeographic areas in function of the overlap threshold (reference is an overlap threshold of ", geographicThresholdVector[2]/2\*100,"%) | For a given species, a biogeographic area difference means that at least one biogeographic area considers absence/presence of the species while this was not the case with the ", geographicThresholdVector[2]/2\*100, "% threshold. ", geographicThresholdVector[2]/2\*100, "% was chosen as the reference since halfway to the chosen maximum of ", geographicThresholdVector[2]\*100, "%. ", geographicThresholdVector[2]\*100, "% was chosen as the maximum because based on current observations, a species occupied at best three different biogeographic areas.", sep="")}

plot(

x=0, y=0, xlab="Overlap threshold", ylab="Variation percent (relative to 15%)",

xlim=c(thresholdPresenceRange[1],thresholdPresenceRange[length(thresholdPresenceRange)]), ylim=c(0,0.4),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=5/100, xmax=30/100, xintsmall=2.5/200, xintbig=5/100, ymin=0, ymax=0.4, yintsmall=0.01, yintbig=0.1, axisPlot=FALSE)

axis(side=2, at=seq(from=0, to=1, by=0.2), labels=seq(from=0, to=1, by=0.2), las=1, tcl=-0.25)

axis(side=1, at=thresholdPresenceRange, labels=thresholdPresenceRange, las=1, tcl=-0.25)

points(x=thresholdPresenceRange, y=howManyDifferent, pch=19, xpd=TRUE)

lines(x=thresholdPresenceRange, y=howManyDifferent)

```

\newpage

### Sensitivity to variation in trait value

```{r figvariabilitydata, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=10, fig.height=4, fig.cap="Supplementary Figure. Variation in trait values among reference datasets | Colours are associated to a specific trait: Brain, Hippocampus, Neocortex and Cerebellum refers to the volume of the area (in mm$^{3}$), Body refers to the body mass (in g), Frug. indicates the frugivory rate and Fol. indicates the folivory rate. (A) Correlation: The points depict the coefficient of correlation while the bar depicts the 95% confidence interval (CI). (B) Variability: The points depict the average of the mean ratio $m$ of the absolute of differences with paired values; If we reduce the equation, we have $m=|(v\_{1}^{2}-v\_{2}^{2})|/(2v\_{1}v\_{2})$, where $v\_{1}$ and $v\_{2}$ are the two paired values from two different datasets and are different from 0. If $v\_{1}$ and $v\_{2}$ equal 0, then $m=0$. If $v\_{1}$ or $v\_{2}$ equals 0 (case for the diet rates constrained between [0,1]), then we fixed the null value to 0.01. The bar depicts the standard error. (C) Repeatability: Repeatability was assessed for traits that were included in at least three datasets. Prior calculation, traits were pondered \*within\* species by the \*within\* species max value. The point represents the mean repeatability $r$ calculated as $V\_{between}/(V\_{between}+V\_{within})$, with the $V\_{between}$ and $V\_{within}$ corresponding the variance \*between\* or \*within\* species. The bar depicts the standard error. For all graphics, sample sizes are indicated above the upper value of the CI/error interval. "}

layout(mat=t(c(1,2,3)), widths=c(5,5,3), heights=c(5))

par(mar=c(4, 3, 2, 1), mgp=c(2, 0.5, 0), xpd=TRUE)

cexText <- c(

rep(0.8, times=10),

rep(0.4, times=3),

rep(0.8, times=6),

rep(0.4, times=3),

rep(1, times=1),

rep(0.8, times=6),

rep(1, times=1),

rep(1, times=1)

)

addToY <- c(

rep(0, times=10),

rep(0, times=3),

rep(0, times=6),

rep(0, times=3),

rep(0, times=1),

rep(0, times=6),

rep(-0.05, times=1),

rep(0, times=1)

)

colourWhatCompared <- c("gray", brewer.pal(n = 9, name = "Pastel1")[c(1:5)], "black", brewer.pal(n = 9, name = "Pastel1")[7])#c(brewer.pal(n = length(unique(whatCompared)) - 1, name = "Pastel1"), "darkgrey")

###------

### CORRELATION

#Vectors to save results

barLower <- rep(NA, times=length(colNumTest))

barUpper <- rep(NA, times=length(colNumTest))

meanCoeff <- rep(NA, times=length(colNumTest))

N <- rep(NA, times=length(colNumTest))

for (i in 1:length(colNumTest)){

test <- #abs(as.numeric(as.character(summaryData[,colNumTest[i]])) - as.numeric(as.character(summaryData[,colNumToCompare[i]])))

cor.test(as.numeric(as.character(summaryData[,colNumTest[i]])),

as.numeric(as.character( summaryData[,colNumToCompare[i]])), method="pearson")

barLower[i] <- test$conf.int[1]

barUpper[i] <- test$conf.int[2]

meanCoeff[i] <- test$estimate[1]

N[i] <- nrow(summaryData[!is.na(summaryData[,colNumTest[i]])&!is.na(summaryData[,colNumToCompare[i]]),])

}

plot(

x=0, y=0, xlab="", ylab="Coefficient of correlation",

xlim=c(0,length(meanCoeff)+1), ylim=c(0.6,1),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=length(meanCoeff), xintsmall=0.5, xintbig=1, ymin=0.6, ymax=1, yintsmall=0.025, yintbig=0.1, axisPlot=FALSE)

axis(side=2, at=seq(from=0.6, to=1, by=0.2), labels=seq(from=0.6, to=1, by=0.2), las=2, tcl=-0.25)

addLabel(x=0.05, y=0.075, label="A", radius=0.75, circle=TRUE, circle.bg="black", font.col="white")

#Comparison

whatCompared <- c(

rep("Brain", times=10),

"Hippocampus",

"Hippocampus",

"Hippocampus",

"Neocortex",

"Neocortex",

"Neocortex",

"Neocortex",

"Neocortex",

"Neocortex",

"Cerebellum",

"Cerebellum",

"Cerebellum",

"Striatum",

rep("Body", times=6),

"Frug.",

"Fol."

)

#Plot legend of what is compared in coloured rectangles

whereToPlot <- as.data.frame(table(whatCompared))

whereToPlot$loc <- whereToPlot$Freq/2

#Colour rectangle to indicate what is compared

refLoc=0

for (i in 1:length(whatCompared)){

rect(xleft=i-1,

xright=i,

ybottom=0.6-0.05\*0.4,#-0.05,

ytop=0.6,#0,

border=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])],

col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])],

xpd=TRUE

)

errorBars(location=i-0.5, meanPt=meanCoeff[i], refUnit=1, col="black", minValue=0, maxValue=1, upperBarValue=barUpper[i], lowerBarValue=barLower[i], horiz=FALSE, symmetrical=FALSE)

points(x=i-0.5, y=meanCoeff[i], pch=19, col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])],

xpd=TRUE)

text(x=i-0.5, y=barUpper[i]+0.015, labels=N[i], pch=19, col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])], cex=0.8,xpd=TRUE)

if(i==length(whatCompared)|whatCompared[i]!=whatCompared[i+1]){

refLoc=refLoc+whereToPlot$loc[whereToPlot[,1]==whatCompared[i]]

if(whereToPlot$loc[whereToPlot[,1]==whatCompared[i]]<=1){

segments(x0=refLoc, x1=refLoc, y0=0.6-0.025\*0.4,#-0.025,

y1=0.6-0.075\*0.4 + addToY[i]\*0.4,#-0.075,

col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])], xpd=TRUE)

text(x=refLoc, y=0.6-0.1\*0.4 + addToY[i]\*0.4,#-0.1,

labels=whereToPlot[whereToPlot[,1]==whatCompared[i],1], col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])], cex=cexText[i], xpd=TRUE)

}

else{

text(x=refLoc, y=0.6-0.025\*0.4,#-0.025,

labels=whereToPlot[whereToPlot[,1]==whatCompared[i],1], col="black", cex=cexText[i], xpd=TRUE)

}

refLoc=refLoc+whereToPlot$loc[whereToPlot[,1]==whatCompared[i]]#add second time for having complete rectangle

}

}

###------

### VARIABILITY

#cbind(colnames(summaryData[colNumTest]), colnames(summaryData[colNumToCompare]))

#Vectors to save results

barLower <- rep(NA, times=length(colNumTest))

barUpper <- rep(NA, times=length(colNumTest))

meanCoeff <- rep(NA, times=length(colNumTest))

N <- rep(NA, times=length(colNumTest))

for (i in 1:length(colNumTest)){

transitoryinit <- as.data.frame(cbind(as.numeric(as.character(summaryData[,colNumTest[i]])),as.numeric(as.character(summaryData[,colNumToCompare[i]]))))

transitoryinit <- transitoryinit[!is.na(transitoryinit[,1])&!is.na(transitoryinit[,2]),]

# transitoryinit <- transitoryinit/max(apply(transitoryinit, 2, max))

transitory <- apply(transitoryinit, 1, function(v){abs(v[1]\*\*2 - v[2]\*\*2)/2/v[1]/v[2]}) #note= (abs((v1-v2))/v1 + abs((v2-v1))/v2)/2

#When rate is 0 for both, gives NA, so to transform to 0

transitory[is.na(transitory)] <- 0

transitory[!is.finite(transitory)] <- apply(transitoryinit[!is.finite(transitory),], 1, function(x) abs(max(x)\*\*2 - 1\*\*2)/2/max(x)/1)

barLower[i] <- mean(transitory) - sd(transitory)/sqrt(length(transitory))#min(transitory)

barUpper[i] <- mean(transitory) + sd(transitory)/sqrt(length(transitory))#max(transitory)

meanCoeff[i] <- mean(transitory)

N[i] <- length(transitory)

}

ymax <- 1#round((barUpper)/10)\*10

plot(

x=0, y=0, xlab="", ylab="Variability",

xlim=c(0,length(meanCoeff)+1), ylim=c(0,ymax),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=length(meanCoeff), xintsmall=0.5, xintbig=1, ymin=0, ymax=ymax, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

axis(side=2, at=seq(from=0, to=ymax, by=0.2), labels=seq(from=0, to=ymax, by=0.2), las=2, tcl=-0.25)

addLabel(x=0.05, y=0.075, label="B", radius=0.75, circle=TRUE, circle.bg="black", font.col="white")

#Comparison

whatCompared <- c(

rep("Brain", times=10),

"Hippocampus",

"Hippocampus",

"Hippocampus",

"Neocortex",

"Neocortex",

"Neocortex",

"Neocortex",

"Neocortex",

"Neocortex",

"Cerebellum",

"Cerebellum",

"Cerebellum",

"Striatum",

rep("Body", times=6),

"Frug.",

"Fol."

)

#Plot legend of what is compared in coloured rectangles

whereToPlot <- as.data.frame(table(whatCompared))

whereToPlot$loc <- whereToPlot$Freq/2

#Colour rectangle to indicate what is compared

refLoc=0

for (i in 1:length(whatCompared)){

rect(xleft=i-1,

xright=i,

ybottom=-0.05,

ytop=0,

border=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])],

col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])],

xpd=TRUE

)

errorBars(location=i-0.5, meanPt=meanCoeff[i], refUnit=1, col="black", minValue=0, maxValue=1, upperBarValue=barUpper[i], lowerBarValue=barLower[i], horiz=FALSE, symmetrical=FALSE)

points(x=i-0.5, y=meanCoeff[i], pch=19, col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])],

xpd=TRUE)

text(x=i-0.5, y=barUpper[i]+0.05, labels=N[i], pch=19, col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])], cex=0.8,xpd=TRUE)

if(i==length(whatCompared)|whatCompared[i]!=whatCompared[i+1]){

refLoc=refLoc+whereToPlot$loc[whereToPlot[,1]==whatCompared[i]]

if(whereToPlot$loc[whereToPlot[,1]==whatCompared[i]]<=1){

segments(x0=refLoc, x1=refLoc, y0=-0.025,

y1=-0.075 + addToY[i],

col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])], xpd=TRUE)

text(x=refLoc, y=-0.1 + addToY[i],

labels=whereToPlot[whereToPlot[,1]==whatCompared[i],1], col=colourWhatCompared[which(unique(whatCompared)==whatCompared[i])], cex=cexText[i], xpd=TRUE)

}

else{

text(x=refLoc, y=-0.025,

labels=whereToPlot[whereToPlot[,1]==whatCompared[i],1], col="black", cex=cexText[i], xpd=TRUE)

}

refLoc=refLoc+whereToPlot$loc[whereToPlot[,1]==whatCompared[i]]#add second time for having complete rectangle

}

}

##----

## Repeatability of measure

colRep <- list(c(4, 14, 17, 22, 23),

c(5, 15, 18),

c(6, 12, 16, 19),

c(7, 13, 20),

#c(8, 21),

c(24, 25, 26, 27)#,

#c(29, 33),

#c(30, 34)

)

ymax <- 1

plot(

x=0, y=0, xlab="", ylab="Repeatability",

xlim=c(0,length(colRep)+1), ylim=c(0,ymax),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

addGrid(xmin=0, xmax=length(colRep), xintsmall=0.5, xintbig=1, ymin=0, ymax=ymax, yintsmall=0.05, yintbig=0.2, axisPlot=FALSE)

axis(side=2, at=seq(from=0, to=ymax, by=0.2), labels=seq(from=0, to=ymax, by=0.2), las=2, tcl=-0.25)

addLabel(x=0.05\*2, y=0.075, label="C", radius=0.75\*length(colRep)/length(meanCoeff)\*2, circle=TRUE, circle.bg="black", font.col="white")

#Comparison

whatCompared2 <- c(

"Brain",

"Hippocampus",

"Neocortex",

"Cerebellum",

#"Striatum",

"Body"#,

#"Fol.",

#"Frug."

)

for(i in 1:length(colRep)){

dataRdc <- summaryData[, colRep[[i]]]

#Normalise by max for all species (i.e. by row)

maxVector <- apply(dataRdc, 1, function(x)max(x, na.rm=TRUE))

maxVector[!is.finite(maxVector)] <- NA

dataRdc <- apply(dataRdc, 2, function(x) x/maxVector)

dataRdc <- as.data.frame(dataRdc)

#Create common ID

dataRdc$id <- 1:nrow(dataRdc)

#Switch to 1 row=1value

dataRdc <- pivot\_longer(dataRdc, col=1:(ncol(dataRdc)-1), names\_to="Dataset", values\_to="Value")

#Remove NAs

dataRdc <- dataRdc[!is.na(dataRdc$Value),]

#Keep thos with multiple obs

whichKeep <- dataRdc %>% count(id)

whichKeep <- whichKeep$id[whichKeep$n > 2]

dataRdc <- dataRdc[dataRdc$id %in% whichKeep,]

dataRdc <- pivot\_wider(dataRdc, names\_from="Dataset", values\_from="Value")

withinVariance <- apply(dataRdc[,2:ncol(dataRdc)], 1, function(x) var(x, na.rm=TRUE))

betweenVariance <- var(apply(dataRdc[,2:ncol(dataRdc)], 1, function(x) mean(x, na.rm=TRUE)), na.rm=TRUE)

repeatability <- betweenVariance/(betweenVariance + withinVariance)

#Plot mean +/- SE

errorBars(location=i-0.5, meanPt=mean(repeatability),

refUnit=1, col="black", minValue=0, maxValue=1, upperBarValue=mean(repeatability)+sd(repeatability)/sqrt(length(repeatability)),

lowerBarValue=mean(repeatability)-sd(repeatability)/sqrt(length(repeatability)), horiz=FALSE, symmetrical=FALSE)

points(x=i-0.5, y=mean(repeatability), pch=19, col=colourWhatCompared[which(unique(whatCompared)==whatCompared2[i])],

xpd=TRUE)

text(x=i-0.5, y=mean(repeatability)+sd(repeatability)/sqrt(length(repeatability))+0.05, labels=length(repeatability), pch=19, col=colourWhatCompared[which(unique(whatCompared)==whatCompared2[i])], cex=0.8, xpd=TRUE)

# library(rptR)

# repeatabilityTest <- rpt(Value ~ (1 | id), grname = "id", data = dataRdc, datatype = "Proportion",

# nboot = 10, npermut = 0)

# Too few data to do that way

}

```

```{r figdata, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=20, fig.height=25, fig.cap="Data availability | Black boxes indicate data availability while grey boxes indicate absence of data."}

#

# plot(

# x=0, y=0, xlab="", ylab="", cex.sub=1.6,

# xlim=c(-10,ncol(dataForSample)-1), ylim=c(0,nrow(dataForSample)),

# las=1, type="n", tcl=-0.25, frame.plot=FALSE,

# xaxt="n",xaxs="i",yaxs="i", yaxt="n")

#

# text(x=rep(-2, times=nrow(dataForSample)), y=1:nrow(dataForSample)-0.5, labels=dataForSample$SpeciesForPhylo, xpd=TRUE, cex=0.4)

# text(x=rep(-0, times=nrow(dataForSample)), y=1:nrow(dataForSample)-0.5, labels=dataForSample$Species, xpd=TRUE, cex=0.4)

# text(x=3:ncol(dataForSample)-1.5, y=rep(nrow(dataForSample)+3, times=length(3:ncol(dataForSample))), labels=colnames(dataForSample)[3:ncol(dataForSample)], xpd=TRUE, cex=0.4, srt=45)

#

# for(i in 1:nrow(dataForSample)){

# for(j in 3:ncol(dataForSample)){

# if(!is.na(dataForSample[i,j])&dataForSample[i,j]==1){

# rect(

# xleft=j-2,

# xright=j-1,

# ybottom=i-1,

# ytop=i,

# border="black",

# col="black"

# )

# } else{

# rect(

# xleft=j-2,

# xright=j-1,

# ybottom=i-1,

# ytop=i,

# border="lightgrey",

# col="lightgrey"

# )

# }

# }

# }

#

# addGrid(xmin=1, xmax=ncol(dataForSample), xintsmall=1, xintbig=1, ymin=0, ymax=nrow(dataForSample), yintsmall=1, yintbig=1, colsmall="white", colbig="white", axisPlot=FALSE)

#

#

#

# plot(

# x=0, y=0, xlab="", ylab="", cex.sub=1.6,

# xlim=c(-10,ncol(dataForSample)-1), ylim=c(0,nrow(dataForSample)),

# las=1, type="n", tcl=-0.25, frame.plot=FALSE,

# xaxt="n",xaxs="i",yaxs="i", yaxt="n")

#

# text(x=rep(-2, times=nrow(dataForSample)), y=1:nrow(dataForSample)-0.5, labels=dataForSample$SpeciesForPhylo, xpd=TRUE, cex=0.4)

# text(x=rep(-0, times=nrow(dataForSample)), y=1:nrow(dataForSample)-0.5, labels=dataForSample$Species, xpd=TRUE, cex=0.4)

# text(x=3:ncol(dataForSample)-1.5, y=rep(nrow(dataForSample)+3, times=length(3:ncol(dataForSample))), labels=colnames(dataForSample)[3:ncol(dataForSample)], xpd=TRUE, cex=0.4, srt=45)

#

# for(i in 1:nrow(dataForSample)){

# for(j in 3:ncol(dataForSample)){

# if(!is.na(dataForSample[i,j])&dataForSample[i,j]==1){

# rect(

# xleft=j-2,

# xright=j-1,

# ybottom=i-1,

# ytop=i,

# border="black",

# col="black"

# )

# } else{

# rect(

# xleft=j-2,

# xright=j-1,

# ybottom=i-1,

# ytop=i,

# border="lightgrey",

# col="lightgrey"

# )

# }

# }

# }

#

# addGrid(xmin=1, xmax=ncol(dataForSample), xintsmall=1, xintbig=1, ymin=0, ymax=nrow(dataForSample), yintsmall=1, yintbig=1, colsmall="white", colbig="white", axisPlot=FALSE)

dataForSample <- dataForSample[dataForSample$SpeciesForPhylo != "Homo\_sapiens",]

dataForSample <- dataForSample[order(dataForSample$SpeciesForPhylo),]

nbPlot=4

layout(mat=t(c(1:nbPlot)), widths=rep(5, times=nbPlot), heights=c(5\*nbPlot))

par(mar=c(0, 0, 3, 0), mgp=c(2, 0.5, 0), xpd=TRUE)

for(p in 1:nbPlot){

dataForSample\_rdcplot <- dataForSample[(1+(p-1)\*nrow(dataForSample)/nbPlot):(1+(p)\*nrow(dataForSample)/nbPlot),]

plot(

x=0, y=0, xlab="", ylab="", cex.sub=1.6,

xlim=c(-10,ncol(dataForSample\_rdcplot)-1), ylim=c(0,nrow(dataForSample\_rdcplot)+3),

las=1, type="n", tcl=-0.25, frame.plot=FALSE,

xaxt="n",xaxs="i",yaxs="i", yaxt="n")

text(x=rep(-5, times=nrow(dataForSample\_rdcplot)), y=1:nrow(dataForSample\_rdcplot)-0.5, labels=dataForSample\_rdcplot$SpeciesForPhylo, xpd=TRUE, cex=1.1)

text(x=rep(-0.5, times=nrow(dataForSample\_rdcplot)), y=1:nrow(dataForSample\_rdcplot)-0.5, labels=dataForSample\_rdcplot$Species, xpd=TRUE, cex=1.1)

text(x=3:ncol(dataForSample\_rdcplot)-1.5, y=rep(nrow(dataForSample\_rdcplot)+2, times=length(3:ncol(dataForSample\_rdcplot))), labels=colnames(dataForSample\_rdcplot)[3:ncol(dataForSample\_rdcplot)], xpd=TRUE, cex=1.1, srt=45)

for(i in 1:nrow(dataForSample\_rdcplot)){

for(j in 3:ncol(dataForSample\_rdcplot)){

if(!is.na(dataForSample\_rdcplot[i,j])&dataForSample\_rdcplot[i,j]==1){

rect(

xleft=j-2,

xright=j-1,

ybottom=i-1,

ytop=i,

border="black",

col="black"

)

} else{

rect(

xleft=j-2,

xright=j-1,

ybottom=i-1,

ytop=i,

border="lightgrey",

col="lightgrey"

)

}

}

}

addGrid(xmin=1, xmax=ncol(dataForSample\_rdcplot), xintsmall=1, xintbig=1, ymin=0, ymax=nrow(dataForSample\_rdcplot), yintsmall=1, yintbig=1, colsmall="white", colbig="white", axisPlot=FALSE)

#if(p==2){cat('\r\n\r\n')} #Allows to consider plot as a new figure at each step of the loop

}

```

\newpage

## Diversification pattern over time

```{r figdiversificationTime, fig.pos='H', include=TRUE, warning = FALSE, message = FALSE, fig.width=5, fig.height=5, fig.cap="Diversification rate over time in the Primate taxon | The average diversification rate estimated based on an assumed sampling fraction of primate species ranging from 60 to 90% (at a step of 10%) is depicted by the plain line. The grey background depicts the standard deviation. The two rupture points, depicted by the plain dots and the vertical dotted bars, were calculated based on a three-linear regression segmentation using the \*strucchange\* package [@strucchange1; @strucchange2; @strucchange3; see the vignette package for statistical details]. The three fitted regressions are displayed by the dashed lines."}

##Plot diversification in function of time

xmin=floor(min(aggregatedSpeciationTime.mean[,1]/10))\*10

xmax=ceiling(max(aggregatedSpeciationTime.mean[,1]/10))\*10

ymin=floor(min(aggregatedSpeciationTime.mean[,2]\*10))/10

ymax=ceiling(max(aggregatedSpeciationTime.mean[,2]\*10))/10

plot(0, 0, xlab="Time before present (Myr)", ylab="Diversification rate",

xlim=c(xmin, xmax), ylim=c(ymin, ymax),

las=1, type="n", tcl=-0.25, bty="n",

xaxt="n",xaxs="i",yaxs="i", yaxt="n",

xpd=TRUE)

#Add grid

addGrid(

xmin=xmin, xmax=xmax, xintsmall=(xmax-xmin)/20, xintbig=(xmax-xmin)/5,

ymin=ymin, ymax=ymax, yintsmall=(ymax-ymin)/20, yintbig=(ymax-ymin)/5,

axisPlot=TRUE, round=TRUE, digit=c(2,2))

axis(side=1, at=round(seq(from=xmin, to=xmax, by=(xmax-xmin)/5), digit=1), labels=round(seq(from=xmin, to=xmax, by=(xmax-xmin)/5), digit=1), las=1, tcl=-0.25)

#Add background se

polygon(

x=c(aggregatedSpeciationTime.mean[,1], rev(aggregatedSpeciationTime.mean[,1])),

y=c(aggregatedSpeciationTime.mean[,2]-aggregatedSpeciationTime.sd[,2], rev(aggregatedSpeciationTime.mean[,2]+aggregatedSpeciationTime.sd[,2])),

col=grey(level=0.5, alpha=0.15),

border=NA

)

#Add mean

lines(aggregatedSpeciationTime.mean[,1], aggregatedSpeciationTime.mean[,2])

library(strucchange)

## confidence interval

colnames(aggregatedSpeciationTime.mean) <- c("Time", "Diversification")

yFirst <- aggregatedSpeciationTime.mean$Diversification[which((abs(aggregatedSpeciationTime.mean$Time+dateFirstRupt[2]))==min(abs(aggregatedSpeciationTime.mean$Time+dateFirstRupt[2])))]

ySecond <- aggregatedSpeciationTime.mean$Diversification[which((abs(aggregatedSpeciationTime.mean$Time+dateSecondRupt[2]))==min(abs(aggregatedSpeciationTime.mean$Time+dateSecondRupt[2])))]

# Rupture points and (CI too reduced to be plotted)

points(c(-dateFirstRupt[2], -dateSecondRupt[2]), c(yFirst, ySecond), pch=19)

# errorBars(location=c(yFirst),

# meanPt=c(-dateFirstRupt[2]),

# barValue=c(0,0), refUnit=1,

# minValue=-80, maxValue=80,

# upperBarValue=c(-dateFirstRupt[1]), lowerBarValue=c(-dateFirstRupt[3]),

# col="black", lty=1,

# horiz=TRUE, symmetrical=FALSE)

#Vertical bars

segments(

x0=c(-dateFirstRupt[2], -dateSecondRupt[2]),

x1=c(-dateFirstRupt[2], -dateSecondRupt[2]),

y0=c(0,0),

y1=c(yFirst, ySecond),

lty=3

)

mtext(side=1, at=c(-dateFirstRupt[2], -dateSecondRupt[2]), line=0, text=c(-dateFirstRupt[2], -dateSecondRupt[2]), cex=0.8)

#three fitted regressions

fm1 <- lm(Diversification ~ breakfactor(bp.resp, breaks = 2)\*Time, data=aggregatedSpeciationTime.mean)

reg1 <- summary(fm1)$coefficients[1,1] + aggregatedSpeciationTime.mean$Time\*summary(fm1)$coefficients[4,1]

reg2 <- summary(fm1)$coefficients[1,1] + summary(fm1)$coefficients[2,1] + aggregatedSpeciationTime.mean$Time\*(summary(fm1)$coefficients[4,1] + summary(fm1)$coefficients[5,1])

reg3 <- summary(fm1)$coefficients[1,1] + summary(fm1)$coefficients[3,1] + aggregatedSpeciationTime.mean$Time\*(summary(fm1)$coefficients[4,1] + summary(fm1)$coefficients[6,1])

lines(aggregatedSpeciationTime.mean$Time, reg1, lty = 2)

lines(aggregatedSpeciationTime.mean$Time[aggregatedSpeciationTime.mean$Time > -40], reg2[aggregatedSpeciationTime.mean$Time > -40], lty = 2)#truncaturate for readability

lines(aggregatedSpeciationTime.mean$Time[aggregatedSpeciationTime.mean$Time > -16], reg3[aggregatedSpeciationTime.mean$Time > -16], lty = 2)#truncaturate for readability

```

## Phylogenetic regressions: results, stability and assumption

### Model results

<!-- We present below the visual fit of phylogenetic regressions. -->

(a) Phylogenetic regressions: selection gradient

```{=latex}

\begin{figure}

\centering\includegraphics[width=0.65\linewidth]{C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Plots/selectionGradientPGLS.pdf}

\caption{\footnotesize{Phylogenetic regressions of the size of the different brain areas in function of the number of sympatric species (left) or the percentage of the range overlapping with the range of other species (right) | The numeric labels refer the brain area number of Figure 1. Left graphics depict the effect of the number of sympatric species on the brain size, when the effect of the percentage of the range overlaped by sympatric species is averaged, while the right graphics do the opposite. Raw data are depicted with points, while the segments that link them correspond to the projected phylogenetic tree. The model fit is shown with the plain black line and the associated 95\% confidence interval is depicted by the transparent gray background.}}\label{fig:figRegressionGradient}

\end{figure}

```

\newpage

(b) Phylogenetic regressions: diversification

```{=latex}

\begin{figure}

\centering\includegraphics[width=0.65\linewidth]{C:/Users/robira/Documents/PhD/Meta\_analysis/Meta\_analysis\_cognition\_primates/Plots/diversificationPGLS.pdf}

\caption{\footnotesize{Phylogenetic regressions of the diversification rate in function of the size of the different brain areas | The numeric labels refer the brain area number of Figure 1. Raw data are depicted with points, while the segments that link them correspond to the projected phylogenetic tree. The model fit is shown with the plain black line and the associated 95\% highest density posterior is depicted by the transparent gray background.}}\label{fig:figRegressionDiversification}

\end{figure}

```

\newpage

### Model stability

We present below statistical indicators related to changes in estimates when re-fitting the model considering sub-samples (i.e. DfBetas and Cook's distance), as well as when accounting for data variability (i.e. re-sampling among possible values given all datasets) or when using different parameterization (i.e. "sampling fraction" of known species for diversification analysis)

```{r correctSensitivityTable}

#Include true estimator in the min-max range

#1)Change min-max interval

summarySensitivityGradient[,1:3+2] <- t(apply(summarySensitivityGradient[,1:3+2], 1, function(x){

x <- as.numeric(x)

if(x[1] > x[2]){

x[1] <- x[2]

} else if (x[2] > x[3]){

x[3] <- x[2]

}else{}

return(x)

}

)

)

summarySensitivityGradient[,4:6+2] <- t(apply(summarySensitivityGradient[,4:6+2], 1, function(x){

x <- as.numeric(x)

if(x[1] > x[2]){

x[1] <- x[2]

} else if (x[2] > x[3]){

x[3] <- x[2]

}else{}

return(x)

}

)

)

#2) Re-round intelligent with scientific writing

summarySensitivityGradient[, 1:6+2] <- t(apply(summarySensitivityGradient[, 1:6+2], 1, function(x)

roundIntelligent(as.numcharac(x))))

#1)Change min-max interval

summarySensitivityDiversification[,1:3+2] <- t(apply(summarySensitivityDiversification[,1:3+2], 1, function(x){

x <- as.numeric(x)

if(x[1] > x[2]){

x[1] <- x[2]

} else if (x[2] > x[3]){

x[3] <- x[2]

}else{}

return(x)

}

)

)

summarySensitivityDiversification[,4:6+2] <- t(apply(summarySensitivityDiversification[,4:6+2], 1, function(x){

x <- as.numeric(x)

if(x[1] > x[2]){

x[1] <- x[2]

} else if (x[2] > x[3]){

x[3] <- x[2]

}else{}

return(x)

}

)

)

summarySensitivityDiversification[,7:9+2] <- t(apply(summarySensitivityDiversification[,7:9+2], 1, function(x){

x <- as.numeric(x)

if(x[1] > x[2]){

x[1] <- x[2]

} else if (x[2] > x[3]){

x[3] <- x[2]

}else{}

return(x)

}

)

)

#2) Re-round intelligent with scientific writing

summarySensitivityDiversification[,-c(1,2)] <- t(apply(summarySensitivityDiversification[,-c(1,2)], 1, function(x) roundIntelligent(as.numcharac(x))))

```

(a) Phylogenetic regressions: selection gradient

```{r tabledfsensitivity, include=TRUE}

knitr::kable(summarySensitivityGradient, escape=TRUE, booktabs = TRUE,

caption = "Sensitivity analysis of phylogenetic regressions to assess the selection gradient direction | Depicted is the minimum and maximum of estimates when one observation was removed at a time (DfBetas) or when varying the used phylogenetic tree and the data sampling (Phylogeny/Data).") %>%

kableExtra::kable\_styling(latex\_options = "striped") %>%

kableExtra::kable\_styling(latex\_options="scale\_down") %>%

kableExtra::kable\_styling(latex\_options = "HOLD\_position") %>%

kableExtra::add\_header\_above(c("Regression" = 2, "DfBetas" = 3, "Phylogeny/Data" = 3))

```

\newpage

(b) Phylogenetic regressions: diversification

```{r tabledfsensitivity2, include=TRUE}

knitr::kable(summarySensitivityDiversification[,-c(3,4,5)], escape=TRUE, booktabs = TRUE,

caption = "Sensitivity analysis of phylogenetic regressions to detect the assess the diversification pattern | Depicted is the minimum and maximum of estimates when varying the used phylogenetic tree and the data sampling (Phylogeny/Data), or when the sampling fraction varied (Sampling fraction).") %>%

kableExtra::kable\_styling(latex\_options = "striped") %>%

kableExtra::kable\_styling(latex\_options="scale\_down") %>%

kableExtra::kable\_styling(latex\_options = "HOLD\_position") %>%

kableExtra::add\_header\_above(c("Regression" = 2, "Phylogeny/Data" = 3, "Sampling fraction" = 3))

```

\newpage

### Model assumptions

We present below the visual assessment of linear modelling assumptions (histogram of residuals, Q-Q plot, and scatterplot of fitted values vs residuals).

(a) Phylogenetic regressions: selection gradient

```{r modelAssumption, include=TRUE, warning = FALSE, message = FALSE, fig.width=7, fig.height=4.5, fig.cap="Model assumption check | Depicted are the histogram of residuals, the Q-Q plot, and the scatter plot of the fitted values vs. the residuals."}

for(a in 1:length(traitName)){

model <- get(paste("modelBrain", traitName[a], sep="\_"))

diagnostics.plot(model)

text(x=0.85, y=0.15, paste("Model:\n",traitName[a], sep=""), xpd=TRUE)

if(a/2==floor(a/2)){

cat('\n') #Break page for new figure

}

}

```

\newpage

(b) Phylogenetic regressions: diversification

```{r modelAssumption2a, include=TRUE, warning = FALSE, message = FALSE, fig.width=7, fig.height=4.5, fig.cap="Trace and density of posteriors"}

for(a in 1:length(traitName)){

model <- get(paste("modelBrainDiversification", traitName[a], sep="\_"))

plot(model1$Sol)

mtext(paste("Fixed effects: ", traitName[a], sep=""), side = 3, line = -1, outer = TRUE, xpd=TRUE)

#cat('\n') #Break page for new figure

plot(model1$VCV)

mtext(paste("Random/residuals: ", traitName[a], sep=""), side = 3, line = -1, outer = TRUE, xpd=TRUE)

#text(x=max(fitted(model)) + abs(min(fitted(model))), y=mean(residuals(model)), paste("Model:\n",traitName[a], sep=""), xpd=TRUE)

if(a/2==floor(a/2)){

cat('\n') #Break page for new figure

}

}

```

```{r modelAssumption2b, include=TRUE, warning = FALSE, message = FALSE, fig.width=7, fig.height=10, fig.cap="Q-Q plot of the posterior distribution and the expected Gaussian distribution"}

layout(mat=matrix(1:(2\*ceiling(length(traitName)/2)), ncol=2), widths=c(5,5), heights=rep(5, times=2\*ceiling(length(traitName)/2)))

par(mar=c(3, 3, 3, 1), mgp=c(2, 0.5, 0), xpd=TRUE)

for(a in 1:length(traitName)){

model <- get(paste("modelBrainDiversification", traitName[a], sep="\_"))

posterior <- as.data.frame(model$Sol)

qqnorm(posterior$Trait, main=paste("Normal Q-Q Plot of the posterior distribution;\nModel:", traitName[a], sep=" "))

qqline(posterior$Trait)

#text(x=max(fitted(model)) + abs(min(fitted(model))), y=mean(residuals(model)), paste("Model:\n",traitName[a], sep=""), xpd=TRUE)

#cat('\n') #Break page for new figure

}

```

\newpage

<!-- TC:endignore -->