**Week 5 Assignment – R Neotoma**

**Exercise Question 1: Use the \*sites\* API to retrieve site data for sites of interest. The \*sites\* API has a few different parameters, so try out options. In your homework exercise, provide at least two \*sites\* API calls with a comment line.**

“[http://api.neotomadb.org/v1/data/sites?sitename=\*Mounds\*](http://api.neotomadb.org/v1/data/sites?sitename=*Mounds*)”

{"success":1,"data":[{"LongitudeWest":-89.866667,"SiteID":281,"SiteName":"Blue Mounds Creek","Altitude":335.0,"LongitudeEast":-89.866667,"SiteDescription":null,"LatitudeNorth":43.083333,"LatitudeSouth":43.083333},{"LongitudeWest":-98.616666666666674,"SiteID":4018,"SiteName":"Jamestown Mounds [32SN22]","Altitude":null,"LongitudeEast":-98.616666666666674,"SiteDescription":null,"LatitudeNorth":46.866666666666674,"LatitudeSouth":46.866666666666674},{"LongitudeWest":-87.5,"SiteID":5291,"SiteName":"Angel Mounds","Altitude":null,"LongitudeEast":-87.5,"SiteDescription":null,"LatitudeNorth":38.0,"LatitudeSouth":38.0},{"LongitudeWest":-93.3977777777778,"SiteID":5491,"SiteName":"Fairfield Mounds [23BE6]","Altitude":null,"LongitudeEast":-93.3977777777778,"SiteDescription":null,"LatitudeNorth":38.138888888888879,"LatitudeSouth":38.138888888888879},{"LongitudeWest":-90.646111111111082,"SiteID":6266,"SiteName":"Elizabeth Mounds","Altitude":null,"LongitudeEast":-90.646111111111082,"SiteDescription":null,"LatitudeNorth":39.68805555555555,"LatitudeSouth":39.68805555555555},{"LongitudeWest":-88.0316666666667,"SiteID":6307,"SiteName":"Wilson Mounds","Altitude":null,"LongitudeEast":-88.0316666666667,"SiteDescription":null,"LatitudeNorth":37.995833333333337,"LatitudeSouth":37.995833333333337},{"LongitudeWest":-90.25,"SiteID":6346,"SiteName":"Dunlieth Mounds","Altitude":null,"LongitudeEast":-90.25,"SiteDescription":null,"LatitudeNorth":42.25,"LatitudeSouth":42.25},{"LongitudeWest":-90.0699,"SiteID":10540,"SiteName":"Cahokia Mounds (Borrow Pit 5-3)","Altitude":126.0,"LongitudeEast":-90.0531,"SiteDescription":"Borrow pit. Physiography: American Bottom. Surrounding vegetation: mixed farmland/prairie.","LatitudeNorth":38.6627,"LatitudeSouth":38.6534},{"LongitudeWest":-90.0699,"SiteID":10541,"SiteName":"Cahokia Mounds (Borrow Pit 5-1)","Altitude":128.0,"LongitudeEast":-90.0531,"SiteDescription":"Borrow pit. Physiography: American Bottom. Surrounding vegetation: mixed farmland/prairie.","LatitudeNorth":38.6627,"LatitudeSouth":38.6534}]}

How to add spaces: (it can also handle spaces if you put astricks on the site name!)

<http://api.neotomadb.org/v1/data/sites?sitename=blue%20mounds%20creek>

“http://api.neotomadb.org/v1/data/sites?sitename=\*Blue Mounds\*”

{"success":1,"data":[{"LongitudeWest":-89.866667,"SiteID":281,"SiteName":"Blue Mounds Creek","Altitude":335.0,"LongitudeEast":-89.866667,"SiteDescription":null,"LatitudeNorth":43.083333,"LatitudeSouth":43.083333}]}

**Exercise Question 2\*\* Do the same for the \*datasets\* and \*downloads\* API (only one API example of each needed). Note that data volumes for objects returned by \*downloads\* can get quite large, so be judicious.**

This is to return a data set that ranges from 250,000 - 5000 years old: <http://api.neotomadb.org/v1/data/datasets?ageold=250000&ageyoung=5000&ageof=dataset>

longitudeWest, latitudeSouth, longitudeEast, latitudeNorth.

<http://api.neotomadb.org/v1/data/datasets?ageold=250000&ageyoung=5000&ageof=dataset>&loc=-92.8,42.4,-87.3,46.9

|  |  |
| --- | --- |
| <http://api.neotomadb.org/v1/data/sites?loc=-92.9,46.3,-77.3,55.2> | Sites within a region in Ontario, Canada |

This is to return datasets with age ranges of 250,000 – 5,000 within the lat lon of the state of WI ( looked up the lat/lon):

http://api.neotomadb.org/v1/data/datasets?ageold=250000&ageyoung=5000&ageof=dataset&loc=-92.8,42.4,-87.3,46.9

**Finding Sites, Getting Metadata: get\_site**

**Exercise Question 3\*\* Write an API call that returns 50 records from the \*Geochronology\* table. (have to use data tables tab)**

http://api.neotomadb.org/v1/dbtables/Geochronology?limit=50

**------**

**print(samwell\_site)**

**site.name long lat elev**

**Samwell Cave -122.2379 40.91691 465**

**A site object containing 1 sites and 8 parameters.**

**Exercise Question 4\*\* How many sites have the name 'clear' in them? Show both your code and provide the total count.**

**clear\_site<-get\_site(sitename='clear%')**

**The API call was successful, you have returned 40 records.**

**Exercise Question 5\*\* Which state has more sites, Minnesota or Wisconsin? How many of each? Provide both code and answer.**

**MN\_sites<-get\_site(gpid="Minnesota")**

The API call was successful, you have returned 518 records.

**WI\_sites <- get\_site(gpid = "Wisconsin")**

The API call was successful, you have returned 459 records.

**Question 6\*\*: How many different kinds of datasets are available at Devil's Lake, WI? Show both code and answer. Ensure that your code just retrieves datasets for just this single site.**

> devil\_sites <- get\_site(sitename = 'devil%')

The API call was successful, you have returned **16 records.**

> devil.meta.dataset <- get\_dataset(devil\_sites)

> devil.meta.dataset

1. A dataset\_list containing **36 objects**: pollen
2. geochronologic
3. pollen
4. pollen surface sample
5. geochronologic
6. water chemistry
7. ostracode surface sample
8. vertebrate fauna
9. vertebrate fauna
10. vertebrate fauna
11. vertebrate fauna
12. vertebrate fauna
13. geochronologic
14. vertebrate fauna
15. vertebrate fauna
16. pollen surface sample
17. geochronologic
18. pollen
19. pollen surface sample
20. water chemistry
21. ostracode surface sample
22. water chemistry
23. ostracode surface sample
24. geochronologic
25. diatom
26. geochronologic
27. pollen
28. water chemistry
29. diatom surface sample
30. geochronologic
31. diatom
32. geochronologic
33. vertebrate fauna
34. diatom
35. diatom
36. vertebrate fauna

**Helper function: *compile\_taxa***

**Make a pollen diagram in *analogue*:**

**analogue-pollen-plot.pdf**

**Question 7: Make a stratigraphic pollen diagram in rioja, for a site of your choice (not Devils lake). Show code and resulting diagram:**

setwd("~/Desktop/Cameron\_Bchron 2/GEOG920/")

library("neotoma")

library("rioja")

bearlake\_datasets<-get\_dataset(10299)

bearlake\_data<-get\_download(bearlake\_datasets)

bearlake\_pollen<-bearlake\_data[[1]]

bearlake\_pollen\_p25<-compile\_taxa(bearlake\_data[[1]],list.name="P25")

#convert pollen species into percents

bearlake\_pollen\_pct<-analogue::tran(x=bearlake\_pollen$counts,method='percent')

#keep only the common taxa (>2%)

bearlake\_pollen\_pct\_norare<-bearlake\_pollen\_pct[,colMeans(bearlake\_pollen\_pct,na.rm=TRUE)>2]

#make row names a numeric value so we can plot depth below

rownames.new<-as.numeric(rownames(bearlake\_pollen\_pct\_norare)

#create a new window

quartz(height=16,width=6)

##defining parameters, but specifically here, number of rows andn columns, margin and outer margin size in inches to reduce white space##

par(mfrow=c(1,1), mai=c(0.1,1,0.25,1),omi=c(0.75,0.1,0.25,0.1))

# Define colour scheme, found this one online and I liked it so kept it.

p.col <- c(rep("forestgreen", times=7), rep("gold2", times=20))

# Define y axis. We want our y-axis to be in depth, so call the 'rownames.new' that I made numerical above

y.scale <- rownames.new

# Plot bar plot using rioja (strat.plot)

pol.plot <- strat.plot(bearlake\_pollen\_pct\_norare, yvar=y.scale, y.tks=y.scale, y.rev=TRUE, plot.line=FALSE, plot.poly=FALSE, plot.bar=TRUE, col.bar=p.col, lwd.bar=10, scale.percent=TRUE, xSpace=0.01, x.pc.lab=TRUE, x.pc.omit0=TRUE, las=2)

#label my axis

xlab="percentages"

**Rioja-plot-question7.pdf**

**Visualization and Analysis: Multiple Sites**

**\*\*Question 8\*\*: Repeat the above exercise, for \*Picea\* or a taxon of your choice. Show code and plot.**

library(neotoma)

pic\_dec<-get\_dataset(taxonname="Picea",datasettype="pollen",loc=c(-98.6,36.5, 66.1,49.75),ageyoung=4500,ageold=6000)

#download all the picea data

pic\_dec\_dl<-get\_download(pic\_dec);

#see where the sites are

library(rworldmap)

map<-getMap()

plot(pic\_dec)

plot(map,add=TRUE)

Map-Picea.pdf

pic\_compiled <- compile\_downloads(pic\_dec\_dl)

# Extract all taxon tables from the original download object

all\_taxa <- do.call(rbind.data.frame, lapply(pic\_dec\_dl, function(x)x$taxon.list[,1:6]))

#Get rid of all the duplicated

all\_taxa <- all\_taxa[!duplicated(all\_taxa),]

good\_cols <- c(1:10, which(colnames(pic\_compiled) %in%

gsub("[ ]|[[:punct:]]", ".",

all\_taxa[all\_taxa$ecological.group %in%

c("TRSH", "UPHE"),1])))

pic\_compiled <- pic\_compiled[ ,good\_cols]

pic\_pct <- pic\_compiled[,11:ncol(pic\_compiled)] / rowSums(pic\_compiled[,11:ncol(pic\_compiled)], na.rm = TRUE)

pic\_only <- rowSums(pic\_pct[,grep("Tsuga", colnames(pic\_pct))], na.rm = TRUE)

#we can pull ages from the compiled\_dowlnoads object (pic\_compiled) b taking the rowMeans of the age columns, and then plot Picea percentages against age

age\_cols<-grep("^age",colnames(pic\_compiled))

picea\_all<-data.frame(ages=rowMeans(pic\_compiled[,age\_cols], na.rm=TRUE),prop=pic\_only)

quartz(height=8,width=6)

par(mfrow=c(3,1), mai=c(0.1,1,0.25,1),omi=c(0.75,0.1,0.25,0.1))

plot(picea\_all,col=rgb(0.1,0.1,0.1,0.3),pch=19,cex=0.4,xlim=c(0,20000),ylab="Proportion of Picea", xlab="Years Before Present")

plot(picea\_all,

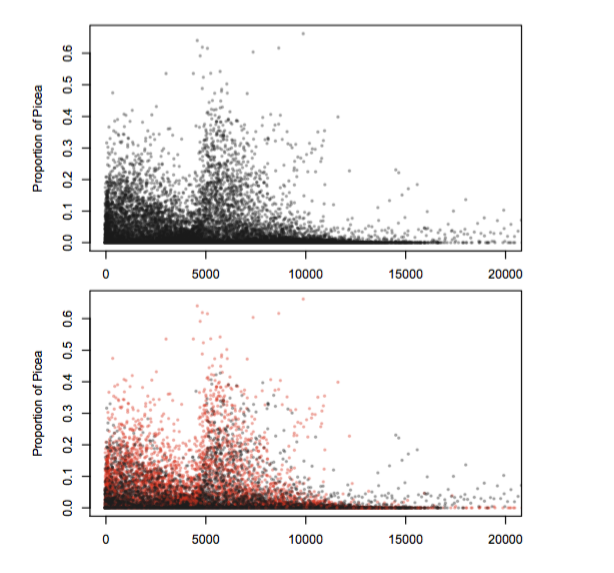
col = c(rgb(0.1, 0.1, 0.1, 0.3),

rgb(1, 0, 0, 0.3))[(pic\_compiled$date.type == "Radiocarbon years BP") + 1],

pch = 19, cex = 0.4,

xlim = c(0, 20000),

ylab = "Proportion of Picea", xlab = "Years Before Present")

****

**Fun with For-Loops**

#----------Repeat the analysis for Picea for the last 21,000 years in the upper midwest.

one\_slice\_new <- get\_dataset(taxonname='Picea\*', loc=c(-150, 20, -100, 60), ageyoung = 0, ageold = 500)

increment <- seq(from = 0, to = 10000, by = 500)

one\_slice\_new <- get\_dataset(taxonname = 'Picea\*', loc = c(-150, 20, -100, 60), ageyoung = increment[1], ageold = increment[2])

for(i in 1:20){ one\_slice\_new <- get\_dataset(taxonname = 'Picea\*', datasettype = 'pollen', loc = c(-150, 20, -100, 60), ageyoung = increment[i], ageold = increment[i + 1]) }

site\_nos\_new <- rep(NA, 20)

for (i in 1:20) { one\_slice\_new <- get\_dataset(taxonname = 'Picea\*', datasettype = 'pollen', loc = c(-150, 20, -100, 60), ageyoung = increment[i], ageold = increment[i + 1])

site\_nos\_new[i] <- length(one\_slice\_new) }

site\_all <- rep(NA, 20)

for (i in 1:20) {

all\_slice <- get\_dataset(datasettype = 'pollen',

loc = c(-150, 20, -100, 60),

ageyoung = increment[i],

ageold = increment[i + 1])

site\_all[i] <- length(all\_slice)

}

quartz(height=16,width=6)

../../../../../../../../Desktop/Cameron_Bchron%202/GEOG920/Picea_pollen_throughtime.pplot(increment[-21], site\_nos/site\_all,xlab="Age (years BP)",ylab="Picea percentage")