Code for 'Stable isotopic evidence of filter feeding mixotrophy in xylophagaids, deep-sea wood-boring bivalves'

Janet Voight, Jacob C. Cooper & Raymond W. Lee

7 November 2019

This document serves to analyze a dataset of deep-water molluscs to ascertain how they differ with respect to several variables. Most importantly, we are trying to determine if populations differ in their amount of $\delta^{15}N$. This code was written by Jacob C. Cooper.

1 Loading the Data

First, we are going to load in the required *R* packages for executing this code:

```
library(tidyverse)
```

```
## -- Attaching packages ----------------------------------------------------------------------- t
```

```
## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
```

```
## -- Conflicts -------------------------------------------------------------------------------- tidyve
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(ranger)
library(phytools)
```

```
## Loading required package: ape
```

```
## Loading required package: maps
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

Next, we are going to load our data file:

```
df=read_csv(paste0(filepath,"voight-data.csv"))
```

```
## Parsed with column specification:
## cols(
##   ID_Number = col_double(),
##   ID_Name = col_character(),
##   Sample = col_character(),
##   C13 = col_double(),
##   N15 = col_double(),
##   Taxon = col_character(),
##   Clade = col_character(),
##   Wood_type = col_character(),
##   Depth = col_double(),
##   Preservation = col_character(),
##   Year = col_double()
## )
```

```
str(df)
```

```
## Classes 'spec_tbl_df', 'tbl_df', 'tbl' and 'data.frame': 72 obs. of  11 variables:
##  $ ID_Number   : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ ID_Name     : chr  "oregana_siphon" "oregana_siphon" "oregana_siphon" "dorsalis-lg_siphon" ...
##  $ Sample      : chr  "oregona siphon 1.raw" "oregona siphon 2.raw" "oregona siphon 3.raw" "dorsalis
##  $ C13         : num  -22.4 -22.7 -22.2 -21.8 -21.8 ...
##  $ N15         : num  1.963 -0.374 2.766 6.14 6.357 ...
##  $ Taxon       : chr  "oregona" "oregona" "oregona" "dorsalis" ...
##  $ Clade       : chr  "dors" "dors" "dors" "dors" ...
##  $ Wood_type   : chr  "fir" "fir" "fir" "wild" ...
##  $ Depth       : num  2211 2211 2211 210 210 ...
##  $ Preservation: chr  "f_to_e" "f_to_e" "f_to_e" "ethanol" ...
##  $ Year        : num  2004 2004 2004 2017 2017 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ID_Number = col_double(),
##   ..   ID_Name = col_character(),
##   ..   Sample = col_character(),
##   ..   C13 = col_double(),
##   ..   N15 = col_double(),
##   ..   Taxon = col_character(),
##   ..   Clade = col_character(),
##   ..   Wood_type = col_character(),
##   ..   Depth = col_double(),
##   ..   Preservation = col_character(),
##   ..   Year = col_double()
##   .. )
```

Each row is a specimen, so the `ID_NAME` column serves to denote species and whether or not the sample was acidified.

2 Random Forest of Variable Importance

Next, we are going to use `randomForest` to look at variable importance for predicting Nitrogen content.

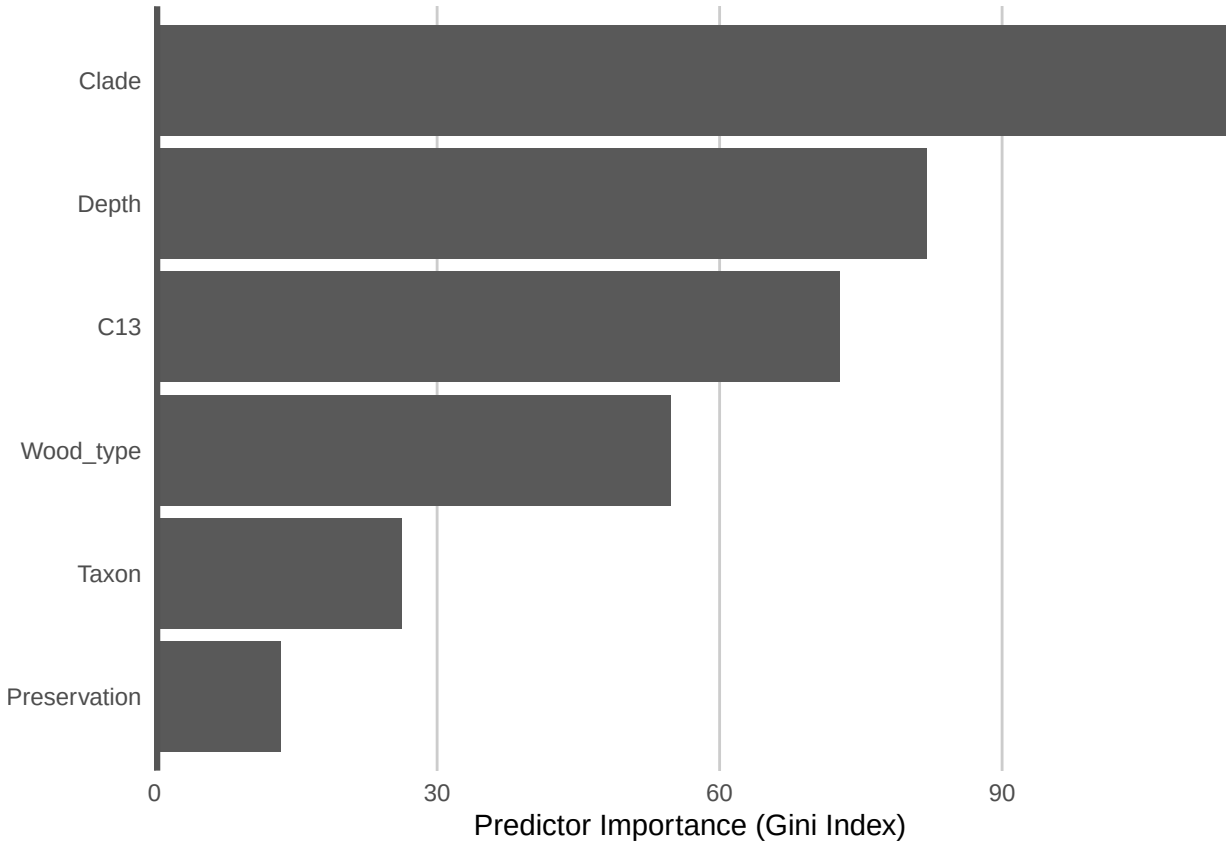
```r
colnames(df)
```

```
##  [1] "ID_Number"    "ID_Name"      "Sample"       "C13"
##  [5] "N15"          "Taxon"        "Clade"        "Wood_type"
##  [9] "Depth"        "Preservation" "Year"
```

```r
df.forest=df %>%
  select(Taxon,C13,N15,Clade,Wood_type,Depth,Preservation)

rf=ranger(formula = N15 ~ .,
          num.trees=1000,
          importance="impurity",
          data=df.forest)

rf
```

```
## Ranger result
##
## Call:
##  ranger(formula = N15 ~ ., num.trees = 1000, importance = "impurity",      data = df.forest)
##
## Type:                             Regression
## Number of trees:                  1000
## Sample size:                      72
## Number of independent variables:  6
## Mtry:                             2
## Target node size:                 5
## Variable importance mode:         impurity
## Splitrule:                        variance
## OOB prediction error (MSE):       1.590023
## R squared (OOB):                  0.7257328
```

```r
pi=enframe(rf$variable.importance, "predictor", "importance")
pi2=pi
# fix names for plots etc.
pi2$predictor=c("C13","Taxon","Clade","Wood_type","Depth","Preservation")
ggplot(pi2)+
  aes(x = fct_reorder(predictor, importance), y = importance) +
  geom_col() +
  geom_hline(yintercept = 0, size = 2, colour = "#555555") +
  scale_y_continuous(expand = c(0, 0)) +
  coord_flip() +
  labs(x = NULL,
       y = "Predictor Importance (Gini Index)",
       fill = "Predictor Importance (Gini Index)") +
  theme_minimal() +
  theme(panel.grid = element_blank(),
        panel.grid.major.x = element_line(colour = "#cccccc", size = 0.5))
```

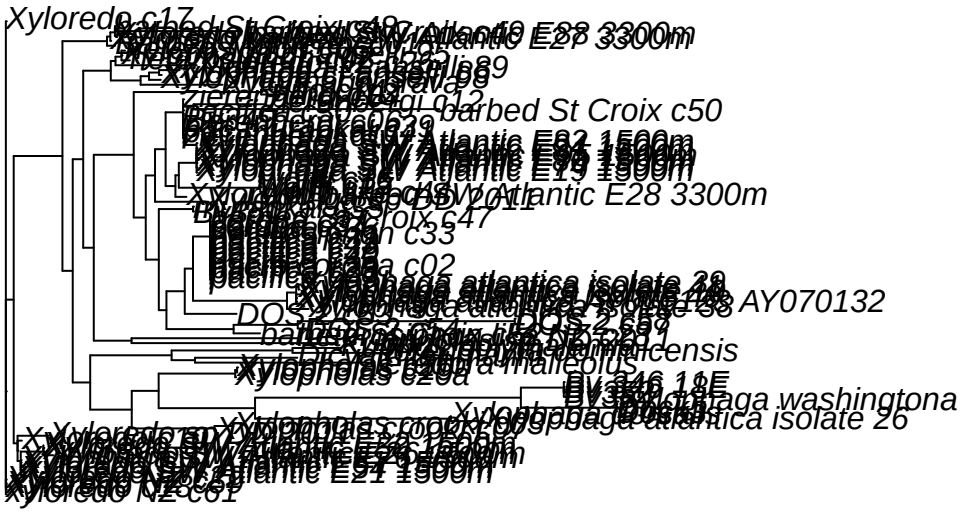


Clade is the most important factor for determining N15, followed by depth. Preservation is the least important metric, and thus we can continue with our analyses under the assumption that we can cross compare all samples. However, we need to be certain that we account for phylogenetic signal within any downstream analyses.

4 Phylogenetically corrected analyses

Now we can load the phylogeny.

```
x.tre=read.tree(paste0(filepath,"RAxML_bipartitions.clams_combined_Dec12-2Xyl_gb.tre"))

plot(x.tre)
```



The tree is a bit messy, but we can see a lot of stuff. We are interested in keeping the tips that contain the names of the species we are studying.

```
x.tre$tip.label
```

```
##  [1] "xyloredo_NZ_c61"
##  [2] "Xyloredo_c18"
##  [3] "Xyloredo_NZ_c59"
##  [4] "Xyloredo_NZ_c22"
##  [5] "Xyloredo_SW_Atlantic_E21_1500m"
##  [6] "Xyloredo_SW_Atlantic_E54_1500m"
##  [7] "Xyloredo_SW_Atlantic_E61_1500m"
##  [8] "Xylophaga_SW_Atlantic_E76_1500m"
##  [9] "Xyloredo_SW_Atlantic_E26_1500m"
## [10] "Xyloredo_SW_Atlantic_E50_1500m"
## [11] "Xyloredo_SW_Atlantic_E22_1500m"
## [12] "Xyloredo_c10"
## [13] "Xyloredo_sp_DD_2011"
## [14] "Xylopholas_crooki_c07"
## [15] "Xylopholas_crooki_c03"
## [16] "Xylophaga_atlantica_isolate_26"
## [17] "Xylophaga_dorsalis"
## [18] "Dock3"
## [19] "Dock5"
## [20] "Xylophaga_washingtona"
## [21] "Bv354"
## [22] "Bv352"
## [23] "Bv_346_18E"
## [24] "Bv_346_11E"
## [25] "Xylopholas_c20a"
## [26] "Xylopholas_c19"
## [27] "Xylopholas_c20b"
## [28] "Teredora_malleolus"
## [29] "Dicyathifer_manni"
## [30] "Teredothyra_dominicensis"
## [31] "Kuphus_polythalamia"
## [32] "Xylopholas_sp_DD_2011"
## [33] "heterosiphon_like_NZ_c28"
```

```
## [34] "barbed_St_Croix_c52"
## [35] "DOS_2_c54"
## [36] "DOS_2_c57"
## [37] "DOS_2_c58"
## [38] "DOS_2_c53"
## [39] "Xylophaga_atlantica_isolate_38"
## [40] "Xylophaga_atlantica_AY070123_AY070132"
## [41] "Xylophaga_atlantica_isolate_40"
## [42] "Xylophaga_atlantica_isolate_19"
## [43] "Xylophaga_atlantica_isolate_41"
## [44] "Xylophaga_atlantica_isolate_39"
## [45] "pacifica_c09"
## [46] "pacifica_c38"
## [47] "pacifica_c29"
## [48] "micro_corona_c02"
## [49] "pacifica_c42"
## [50] "pacifica_c40"
## [51] "pacifica_c32"
## [52] "pacifica_c43"
## [53] "pacifica_c31"
## [54] "heterosiphon_c33"
## [55] "pacifica_c36"
## [56] "corona_c34"
## [57] "corona_c37"
## [58] "barbed_St_Croix_c47"
## [59] "Bv351"
## [60] "muraokai_c35"
## [61] "Xylophaga_sp_DD_2011"
## [62] "Xyloredo_barbed_SW_Atlantic_E28_3300m"
## [63] "wolffi_like_c44"
## [64] "wolffi_c27"
## [65] "wolffi_c15"
## [66] "wolffi_c16"
## [67] "Xylophaga_SW_Atlantic_E19_1500m"
## [68] "Xylophaga_SW_Atlantic_E74_1500m"
## [69] "Xylophaga_SW_Atlantic_E80_1500m"
## [70] "Xylophaga_SW_Atlantic_E95_1500m"
## [71] "Xylophaga_SW_Atlantic_E81_1500m"
## [72] "Xylophaga_SW_Atlantic_E94_1500m"
## [73] "Xylophaga_SW_Atlantic_E82_1500m"
## [74] "zierenbergi_c11"
## [75] "pac_muraokai_c41"
## [76] "pac_muraokai_c39"
## [77] "zierenbergi_c06"
## [78] "Bv349"
## [79] "pacifica_c30"
## [80] "barbed_St_Croix_c50"
## [81] "zierenbergi_c12"
## [82] "zierenbergi_c04"
## [83] "uno_c13"
## [84] "uno_c14"
## [85] "Xylophaga_brava"
## [86] "Xylophaga_cf_anselli_98"
## [87] "Xylophaga_cf_anselli_86"
```

```
## [88] "Xylophaga_cf_anselli_89"
## [89] "Xyloredo_NZ_c26"
## [90] "heterosiphon_c08"
## [91] "Xylophaga_cf_anselli_95"
## [92] "heterosiphon_c05"
## [93] "Barbed_Monterey_c01"
## [94] "Xyloredo_barbed_SW_Atlantic_E27_3300m"
## [95] "Xyloredo_barbed_SW_Atlantic_E88_3300m"
## [96] "barbed_St_Croix_c49"
## [97] "barbed_St_Croix_c48"
## [98] "Xyloredo_c17"
```

```
taxa=unique(df$Taxon)
taxa
```

```
##  [1] "oregona"     "dorsalis"    "washingtona" "crooki"      "nooi"
##  [6] "alexisi"     "microchira"  "heterosiph"  "muraokai"    "zierenber"
```

```
x.matches=NA

#i=2

for(i in 1:length(taxa)){
  name=taxa[i]
  index=which(x.tre$tip.label %like% name)
  if(length(index)==0){
    print(paste0("No matches available. Find: ",name))
  }else{
    if(is.na(matches)){
      x.matches=cbind(name,index)
    }else{
      matches2=cbind(name,index)
      x.matches=rbind(x.matches,matches2)
    }
  }
}
```

```
## [1] "No matches available. Find: oregona"
## Warning in is.na(matches): is.na() applied to non-(list or vector) of type
## 'closure'

## Warning in is.na(matches): is.na() applied to non-(list or vector) of type
## 'closure'

## Warning in is.na(matches): is.na() applied to non-(list or vector) of type
## 'closure'
## [1] "No matches available. Find: nooi"
## [1] "No matches available. Find: alexisi"
## [1] "No matches available. Find: microchira"
## Warning in is.na(matches): is.na() applied to non-(list or vector) of type
## 'closure'

## Warning in is.na(matches): is.na() applied to non-(list or vector) of type
## 'closure'
```

```
## Warning in is.na(matches): is.na() applied to non-(list or vector) of type
## 'closure'
```

```
x.matches=as.data.frame(na.omit(x.matches))
x.matches
```

```
##             name index
## X       dorsalis    17
## X.1  washingtona    20
## X.2       crooki    14
## X.3       crooki    15
## X.4   heterosiph    33
## X.5   heterosiph    54
## X.6   heterosiph    90
## X.7   heterosiph    92
## X.8     muraokai    60
## X.9     muraokai    75
## X.10    muraokai    76
## X.11   zierenber    74
## X.12   zierenber    77
## X.13   zierenber    81
## X.14   zierenber    82
```

We still need to find *oregana*, *nooi*, *alexisi*, and *microchira*. We can double check our matches:

```
# check matches
for(i in 1:nrow(x.matches)){
  n=as.numeric(as.character(x.matches$index[i]))
  print(paste0(x.matches$name[i]," :: ",x.tre$tip.label[n]))
}
```

```
## [1] "dorsalis :: Xylophaga_dorsalis"
## [1] "washingtona :: Xylophaga_washingtona"
## [1] "crooki :: Xylopholas_crooki_c07"
## [1] "crooki :: Xylopholas_crooki_c03"
## [1] "heterosiph :: heterosiphon_like_NZ_c28"
## [1] "heterosiph :: heterosiphon_c33"
## [1] "heterosiph :: heterosiphon_c08"
## [1] "heterosiph :: heterosiphon_c05"
## [1] "muraokai :: muraokai_c35"
## [1] "muraokai :: pac_muraokai_c41"
## [1] "muraokai :: pac_muraokai_c39"
## [1] "zierenber :: zierenbergi_c11"
## [1] "zierenber :: zierenbergi_c06"
## [1] "zierenber :: zierenbergi_c12"
## [1] "zierenber :: zierenbergi_c04"
```

```
# make sure everything is formatted correctly

x.matches$index=as.numeric(as.character(x.matches$index))
```

We have the added additional information:

- *nooi* refers to the genus *Xyloredo*, which only has two sister species and is in the aforementioned phylogeny.

still need to find *oregana alexisi*, and *microchira*.

Now we can reduce the tree.

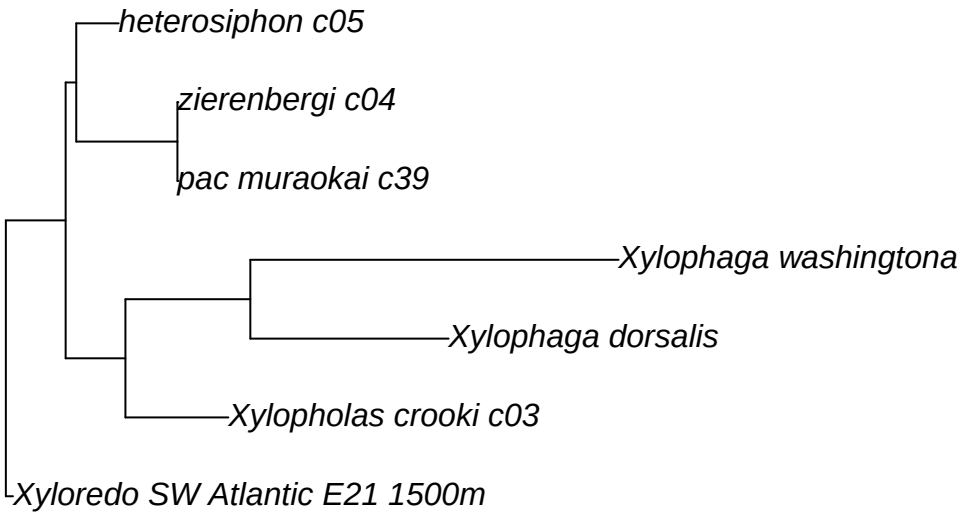
```
index=which(x.tre$tip.label %like% "Xyloredo")

name="nooi"
xx.match=as.data.frame(cbind(name, index))

xx.match$index=as.numeric(as.character(xx.match$index))

x.matches=rbind(x.matches,xx.match) %>% unique()

tips=x.matches$index

x.tre2=keep.tip(x.tre,tips)
plot(x.tre2)
```

We still have a lot of tips. We want one distance for each species, since this is a different dataset that what we are using.

```
names=x.matches$name %>% unique()

for(i in 1:length(names)){
  index=as.numeric(x.matches$index[which(x.matches$name %like% names[i])])
  if(length(index)==1){x=index}else(x=sample(index,1))
  if(i==1){tips=x}else(tips=c(tips,x))
}

tips=tips %>% unique()

names
```

```
## [1] dorsalis    washingtona crooki      heterosiph  muraokai    zierenber
## [7] nooi
## 7 Levels: crooki dorsalis heterosiph muraokai washingtona ... nooi
```

```
x.tre2=keep.tip(x.tre,tips)
plot(x.tre2)
```



Now to reduce the dataset by the same species.

```
for(i in 1:length(names)){
  index=which(df$Taxon==names[i])
  if(i==1){subset=index}else{subset=c(subset,index)}
}

df.anova=df[subset,]
unique(df.anova$Taxon)
```

```
## [1] "dorsalis"    "washingtona" "crooki"      "heterosiph"  "muraokai"
## [6] "zierenber"   "nooi"
```

Now we can perform the phylANOVA.

```
x.tre2$tip.label
```

```
## [1] "Xyloredo_SW_Atlantic_E21_1500m" "Xylopholas_crooki_c03"
## [3] "Xylophaga_dorsalis"             "Xylophaga_washingtona"
## [5] "pac_muraokai_c39"               "zierenbergi_c04"
## [7] "heterosiphon_c05"
```

```
for(i in 1:length(x.tre2$tip.label)){
  if((x.tre2$tip.label[i] %like% "Xyloredo")==T){x.tre2$tip.label[i]="nooi"}
  if((x.tre2$tip.label[i] %like% "crooki")==T){x.tre2$tip.label[i]="crooki"}
  if((x.tre2$tip.label[i] %like% "dorsalis")==T){x.tre2$tip.label[i]="dorsalis"}
  if((x.tre2$tip.label[i] %like% "washingtona")==T){x.tre2$tip.label[i]="washingtona"}
  if((x.tre2$tip.label[i] %like% "heterosiph")==T){x.tre2$tip.label[i]="heterosiph"}
  if((x.tre2$tip.label[i] %like% "muraokai")==T){x.tre2$tip.label[i]="muraokai"}
  if((x.tre2$tip.label[i] %like% "zierenber")==T){x.tre2$tip.label[i]="zierenber"}
}

x.tre2$tip.label
```

```
## [1] "nooi"        "crooki"      "dorsalis"    "washingtona" "muraokai"
## [6] "zierenber"   "heterosiph"
```

```
x.names=df.anova$Taxon
x.groups=df.anova$Clade
y.data=df.anova$C13
```

```r
names(y.data)=x.names
names(x.groups)=x.names

phylANOVA(x.tre2,x.groups,y.data)
```

```
## ANOVA table: Phylogenetic ANOVA
##
## Response: y
##             Sum Sq  Mean Sq  F value Pr(>F)
## x        16.400238 4.100060 7.458129  0.305
## Residual  1.099487 0.549744
##
## P-value based on simulation.
## ---------
##
## Pairwise posthoc test using method = "holm"
##
## Pairwise t-values:
##        abdito dors pholas redo xylo
## abdito     NA   NA     NA   NA   NA
## dors       NA   NA     NA   NA   NA
## pholas     NA   NA     NA   NA   NA
## redo       NA   NA     NA   NA   NA
## xylo       NA   NA     NA   NA   NA
##
## Pairwise corrected P-values:
##        abdito dors pholas redo xylo
## abdito     NA   NA     NA   NA   NA
## dors       NA   NA     NA   NA   NA
## pholas     NA   NA     NA   NA   NA
## redo       NA   NA     NA   NA   NA
## xylo       NA   NA     NA   NA   NA
## ---------
```

```r
y.data=df.anova$N15
names(y.data)=x.names

phylANOVA(x.tre2,x.groups,y.data)
```

```
## ANOVA table: Phylogenetic ANOVA
##
## Response: y
##             Sum Sq   Mean Sq   F value Pr(>F)
## x        43.426563 10.856641 11.623705   0.23
## Residual  1.868017  0.934009
##
## P-value based on simulation.
## ---------
##
## Pairwise posthoc test using method = "holm"
##
## Pairwise t-values:
##        abdito dors pholas redo xylo
## abdito     NA   NA     NA   NA   NA
```

```
## dors       NA   NA     NA   NA   NA
## pholas     NA   NA     NA   NA   NA
## redo       NA   NA     NA   NA   NA
## xylo       NA   NA     NA   NA   NA
##
## Pairwise corrected P-values:
##        abdito dors pholas redo xylo
## abdito     NA   NA     NA   NA   NA
## dors       NA   NA     NA   NA   NA
## pholas     NA   NA     NA   NA   NA
## redo       NA   NA     NA   NA   NA
## xylo       NA   NA     NA   NA   NA
## ---------
```

We do not find a significant effect of either Carbon or Nitrogen in relation to phylogenetic history. This is true during repetitions as well - we do not find a significant effect of clade.

5 Regular ANOVA & Kruskal-Wallis analyses

Now we can also perform regular ANOVA analyses on these data.

```
df.x=df %>%
  select(C13,N15,Taxon,Clade,Wood_type,Depth,Preservation)

summary(as.factor(df.x$Taxon))
```

```
##     alexisi      crooki    dorsalis  heterosiph  microchira    muraokai
##           4           3           2           4           4           5
##        nooi     oregona washingtona   zierenber
##           1           3           4          42
```

We have very small groups for all except for one individual. This makes it non-parametric, especially since we have only one individual. We should use Kruskal-Wallis tests to address this.

```
kt.df=kruskal.test(N15~Taxon,df.x)

kt.df
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  N15 by Taxon
## Kruskal-Wallis chi-squared = 44.009, df = 9, p-value = 1.406e-06
```

There is a significant effect of Taxon on the data, which is unsurprising given that many groups are one species, and certain species occur in certain places.

```
kt.df=kruskal.test(N15~Clade,df.x)

kt.df
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  N15 by Clade
## Kruskal-Wallis chi-squared = 35.145, df = 4, p-value = 4.337e-07
```

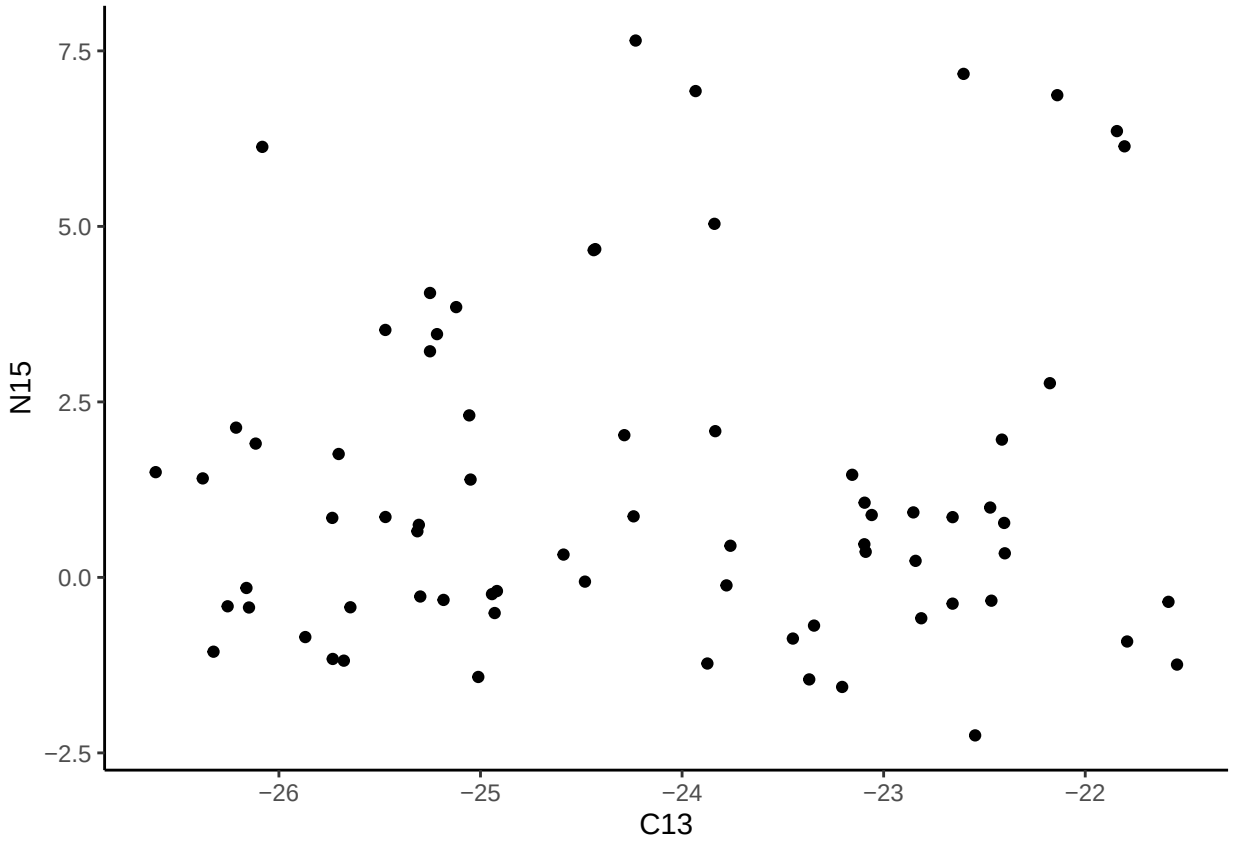
In this case, we find a significant effect for clade, but we have already shown from the phylogenetically corrected test that this is not significant.

```
kt.df=kruskal.test(N15~C13,df.x)

kt.df
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  N15 by C13
## Kruskal-Wallis chi-squared = 71, df = 71, p-value = 0.4777
```

There is no significant effect of carbon on the Nitrogen level.

```
ggplot(df.x,aes(x=C13,y=N15))+geom_point()+theme_classic()
```




```
df.x$Wood_type[which(df.x$Wood_type=="wild")]=NA

df.wood=df.x %>% na.omit()

kt.df=kruskal.test(N15~Wood_type,df.wood)

kt.df
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  N15 by Wood_type
## Kruskal-Wallis chi-squared = 26.809, df = 6, p-value = 0.0001572
```

There is a significant effect of wood type on Nitrogen level.

How does wood type relate to taxon?

```
table(df.x$Taxon,df.x$Wood_type)
```

```
##
##               deployed fir ginkgo ironwood oak pine spicebuch
##   alexisi            4   0      0        0   0    0         0
##   crooki             0   0      0        0   0    0         0
##   dorsalis           0   0      0        0   0    0         0
##   heterosiph         0   4      0        0   0    0         0
##   microchira         0   4      0        0   0    0         0
##   muraokai           0   2      3        0   0    0         0
##   nooi               0   0      0        0   0    0         0
##   oregona            0   3      0        0   0    0         0
##   washingtona        0   0      0        0   0    0         0
##   zierenber          0   2      9        3  10    7        11
```

Almost all this variation is within *zierenbergeri*, and thus probably is a real effect of the wood type.

```
kt.df=kruskal.test(N15~Preservation,df.x)

kt.df
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  N15 by Preservation
## Kruskal-Wallis chi-squared = 7.7495, df = 1, p-value = 0.005373
```

There is a significant effect of preservation type.

```
table(df.x$Taxon,df.x$Preservation)
```

```
##
##               ethanol f_to_e
##   alexisi           0      4
##   crooki            0      3
##   dorsalis          2      0
##   heterosiph        0      4
##   microchira        0      4
##   muraokai          3      2
##   nooi              0      1
##   oregona           0      3
##   washingtona       4      0
##   zierenber        40      2
```

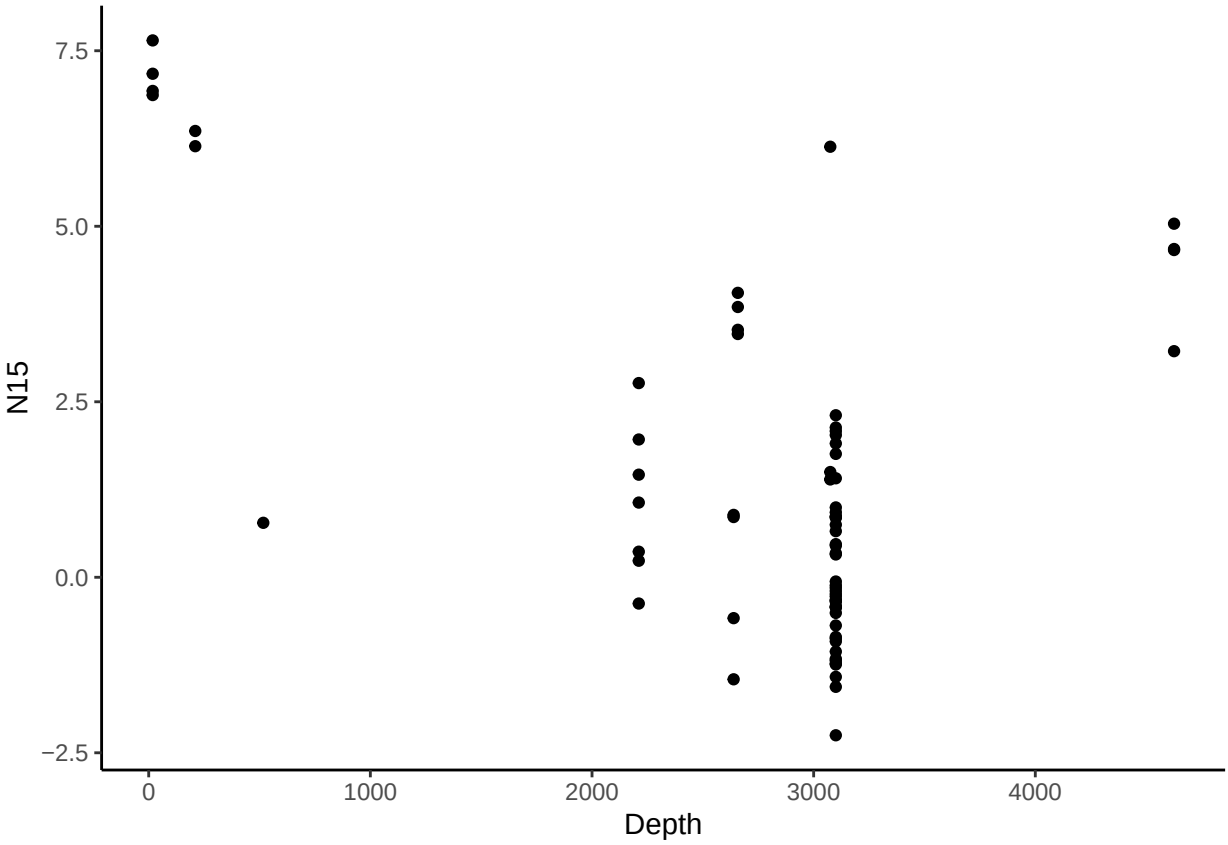
Again, this may be confounded by species, since some species were only preserved one way.

```
kt.df=kruskal.test(N15~Depth,df.x)

kt.df
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  N15 by Depth
## Kruskal-Wallis chi-squared = 39.691, df = 8, p-value = 3.657e-06
```

Depth is a significant effect for N17.

```
ggplot(df.x,aes(x=Depth,y=N15))+geom_point()+theme_classic()
```



Literature Cited

Random forest processing borrows parts of the following pipeline for analysis and visualization:

Johnston, A, WM Hochachka, ME Strimas-Mackey, V Ruiz Gutierrez, OJ Robinson, ET Miller, T Auer, ST Kelling & D Fink. BioRXiv Preprint. Best practices for making reliable inferences from citizen science data: case study using eBird to estimate species distributions.

Taxonomic data are from the following study:

Voight, JR, BA Marshall, J Judge, KM Halanych, Y Li, AF Bernardino, F Grewe & JD Maddox. 2019. Life in wood: preliminary phylogeny of deep-sea wood-boring bivalves (Xylophagaidae), with descriptions of three new genera and one new species. *J. Moll. Stud.* 85, 232-243. doi:10.1093/mollus/eyz003

Citations for packages used in the creation of this document and pipeline are below:

Allaire, JJ, Y Xie, J McPherson, J Luraschi, K Ushey, A Atkins, H Wickham, J Cheng, W Chang & R Iannone (2019). rmarkdown: Dynamic Documents for R. R package version 1.16.

Dowle, M & A Srinivasan (2019). data.table: Extension of `data.frame`. R package version 1.12.6.

R Core Team (2019). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Revell, LJ (2012) phytools: An R package for phylogenetic comparative biology (and other things). *Methods Ecol. Evol.* 3, 217-223. doi:10.1111/j.2041-210X.2011.00169.x

Wickham, H (2017). tidyverse: Easily Install and Load the 'Tidyverse'. R package version 1.2.1.

Wright, MN & A Ziegler (2017). ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. *J of Statistical Software* 77(1), 1-17. doi:10.18637/jss.v077.i01

Xie, Y, JJ Allaire & G Grolemund (2018). R Markdown: The Definitive Guide. Chapman and Hall/CRC. ISBN 9781138359338.