Assigning simple morphologies

Fraass

7/19/2018

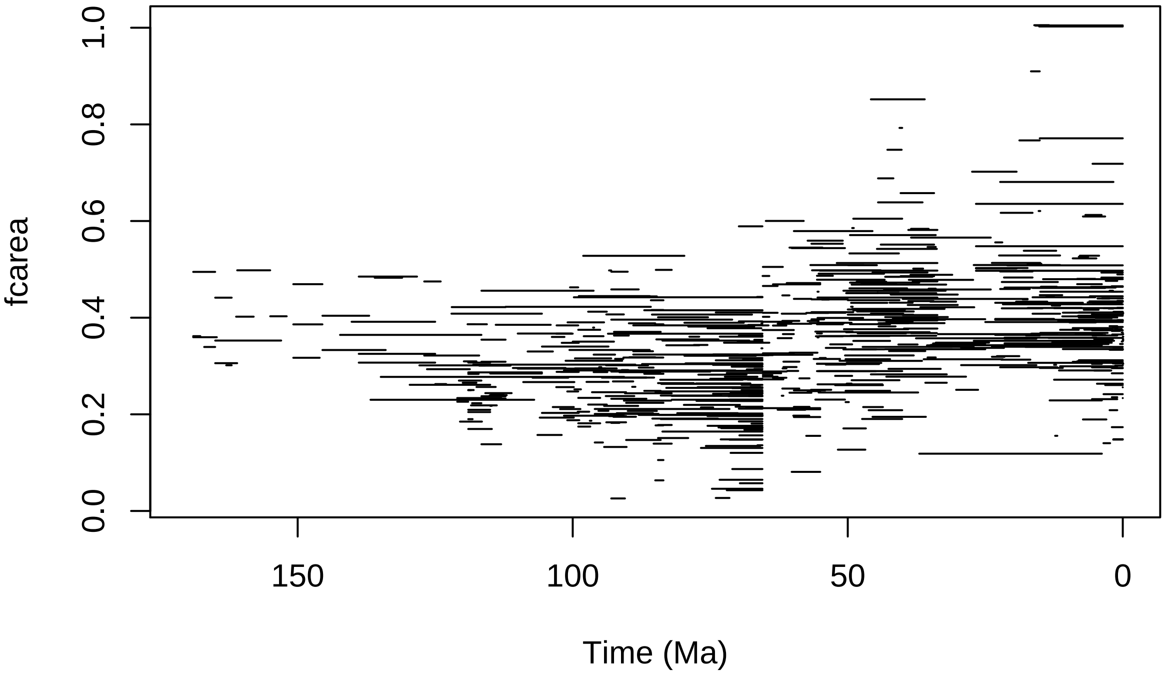
As I sit here this morning, I'm reminded of the fact that this is exactly how we started this project. We've literally come back around to picking the two simplest bugs and doing the analysis that way. That's *exactly* how we started.

OH WELL, on with the show.

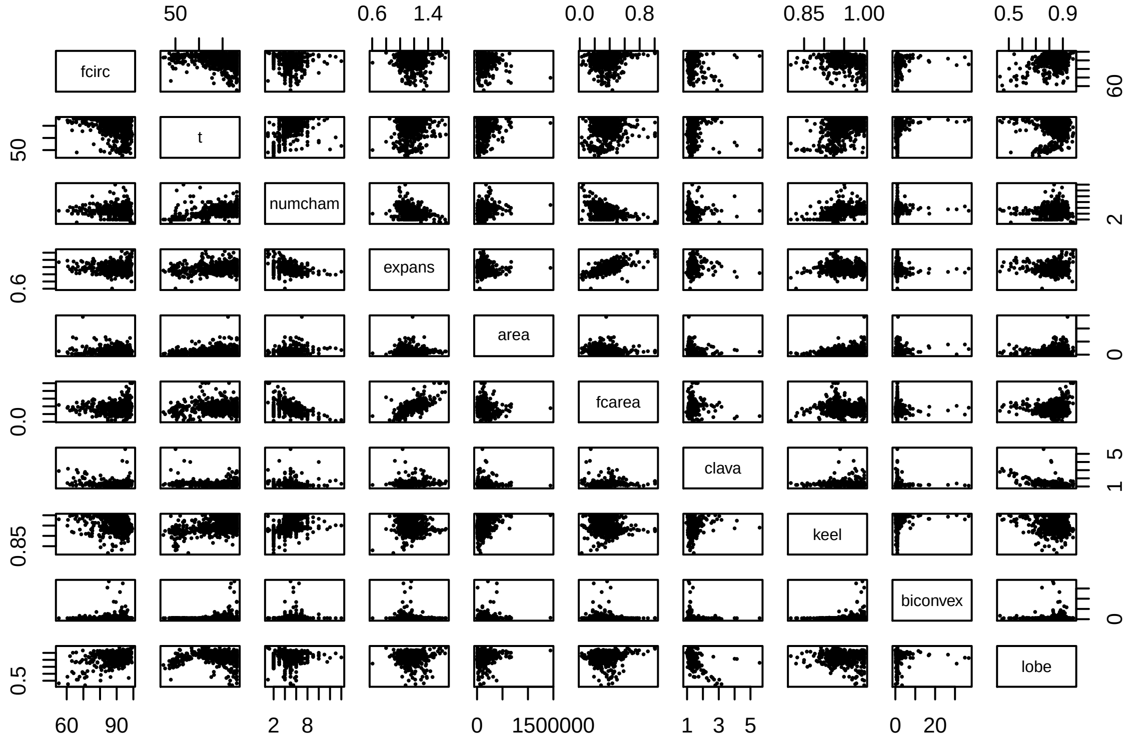
```
read.csv(file="/Users/andrewfraass/Science!/Research/MacroEvo/Data & Analysis/dataframes/MasterList (vs
foram.dataframe->morph

"%w/o%" <- function(x, y) x[!x %in% y] #--  x without y

exclude<-c(4)
morph[1:length(morph[,1] %w/o% exclude),]->morph
```

This adds the files, excluding the one that is missing data.

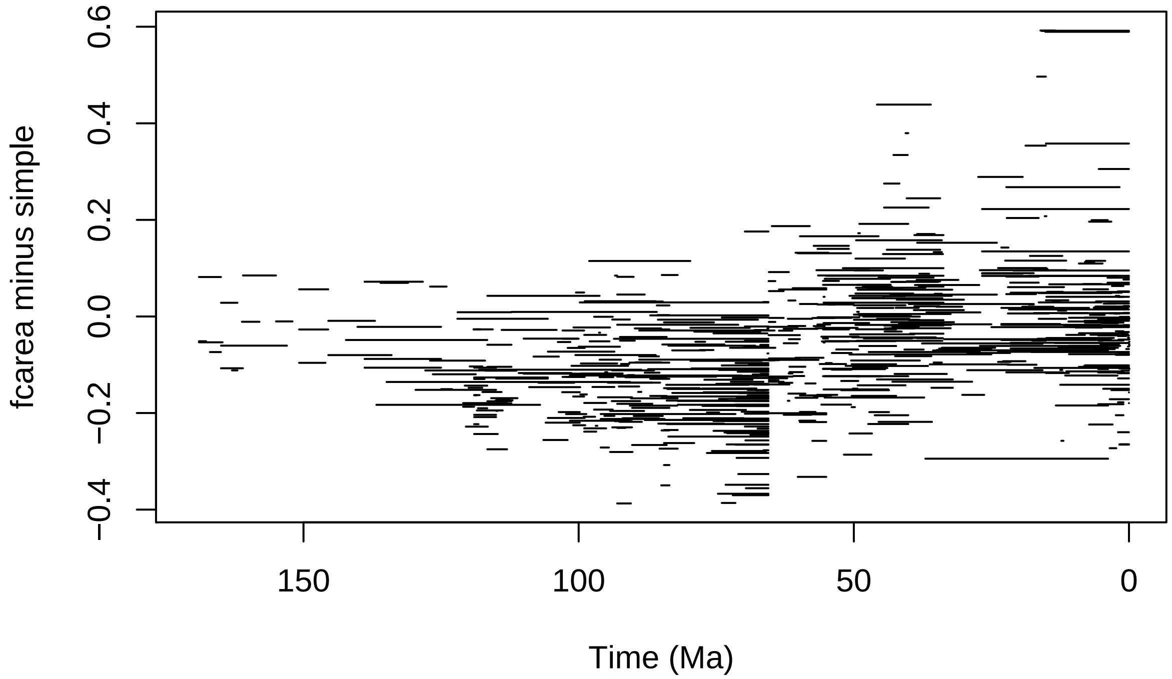
```
#basic plot
plot(0,0,
     type='n',
     xlim=c(170,0),
     ylim=c(min(foram.dataframe$fcarea,na.rm=T),
            max(foram.dataframe$fcarea,na.rm=T)),
     xlab='Time (Ma)',
     ylab="fcarea")
segments(foram.dataframe$origin,
         foram.dataframe$fcarea,
         foram.dataframe$extin,
         foram.dataframe$fcarea)
```



```r
inc<-c(#"w",
  #"mtheta",
  #"lw",
  #"lh",
  #"ic1",
  #"ic2",
  #"ic3",
  "fcirc",
  "t",
  "numcham",
  "expans",
  #"height",
  #"length",
  #"fcangle",
  "area",
  "fcarea",
  "clava",
  #"chamwl",
  "keel",
  #"bidors",
  #"biven",
  "biconvex",
  "lobe"
  #"double"
  #"depth"
)
pairs(morph[,inc],pch=16,cex=.4)
```

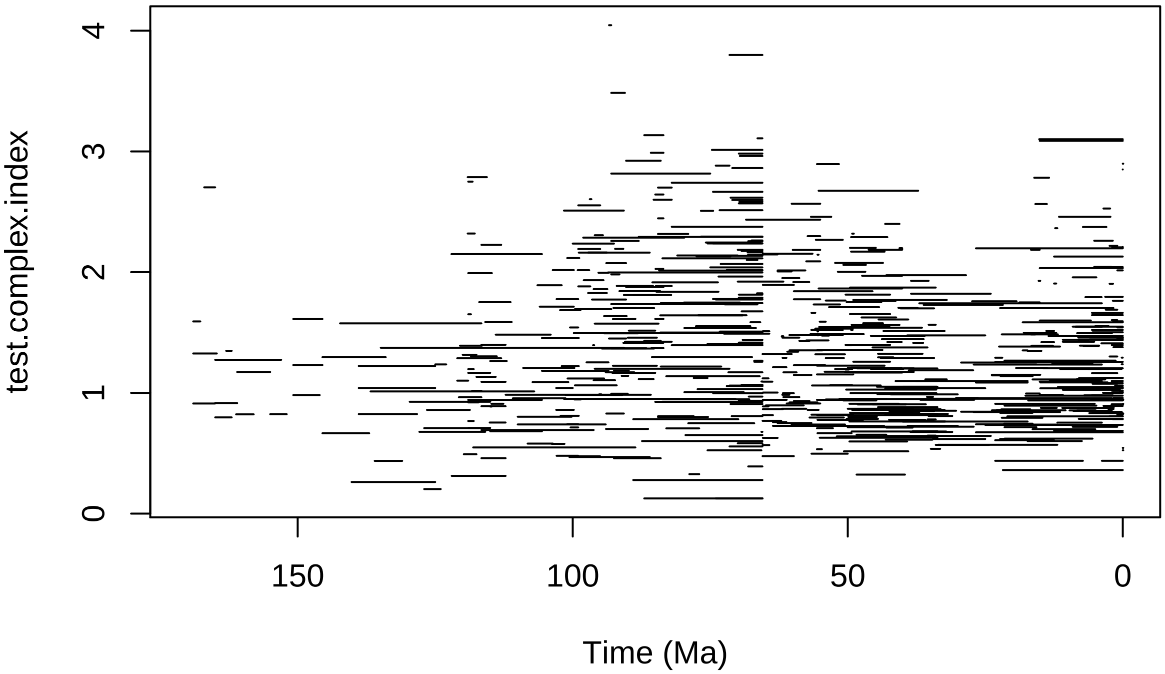


```
which(morph$species == "holmdelensis")->h
which(morph$species == "monmouthensis")->m
build.tci<-morph
for(i in inc){
  {morph[h,i]+morph[m,i]}/2->simple.value
  build.tci[,i]<-morph[,i]-simple.value
}
plot(0,0,
     type='n',
     xlim=c(170,0),
     ylim=c(min(build.tci$fcarea,na.rm=T),
            max(build.tci$fcarea,na.rm=T)),
     xlab='Time (Ma)',
     ylab="fcarea minus simple")
segments(build.tci$origin,
         build.tci$fcarea,
         build.tci$extin,
         build.tci$fcarea)
```



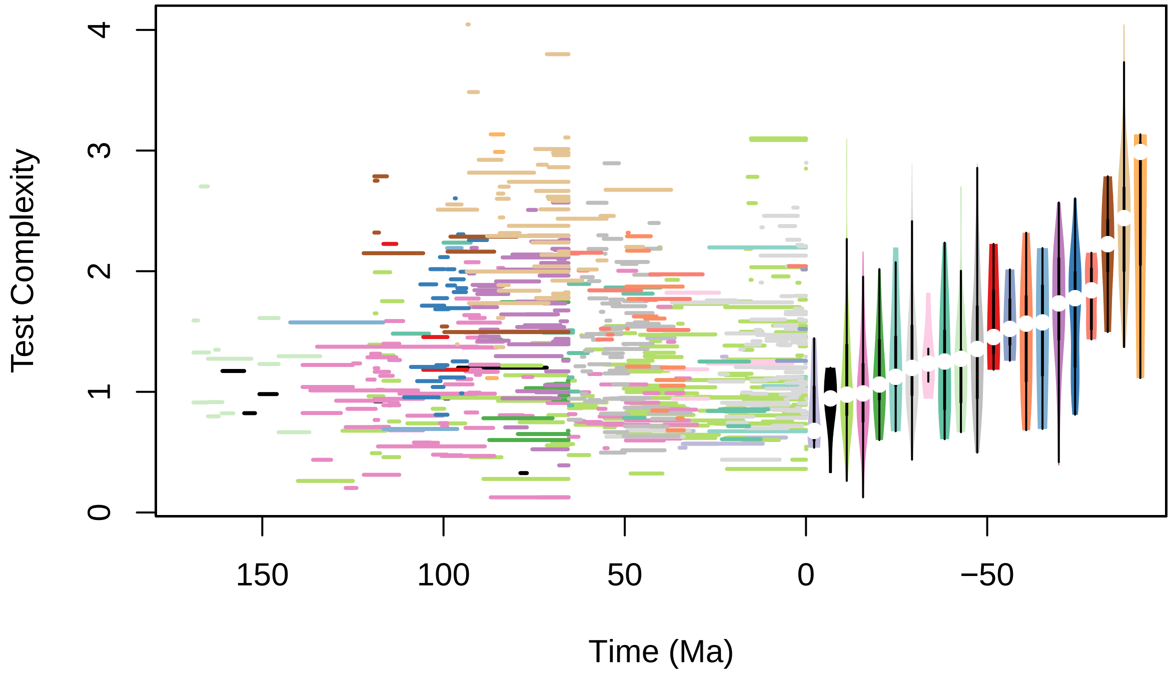
So here's a plot showing the a issue with picking one end. there are changes in the final chamber area/test area that are highly different from Meso->Ceno. Which is odd, but it's probably hte orbulinids and others like it.Either way, our assigned simple bugs are smack in the middle. Neat how it drifts one direction and the next.

```
for(i in inc){
  abs(build.tci[,i])->build.tci[,i]#taking the absolute value
  decostand(build.tci[,i],method='range',na.rm=TRUE)->build.tci[,i]
}
test.complex.index<-NA
for(i in 1:length(build.tci$species)){
  sum(build.tci[i,inc])->test.complex.index[i]
}
plot(0,0,
     type='n',
     xlim=c(170,0),
     ylim=c(min(test.complex.index,na.rm=T),
            max(test.complex.index,na.rm=T)),
     xlab='Time (Ma)',
     ylab="test.complex.index")
segments(build.tci$origin,
         test.complex.index,
         build.tci$extin,
         test.complex.index)
```



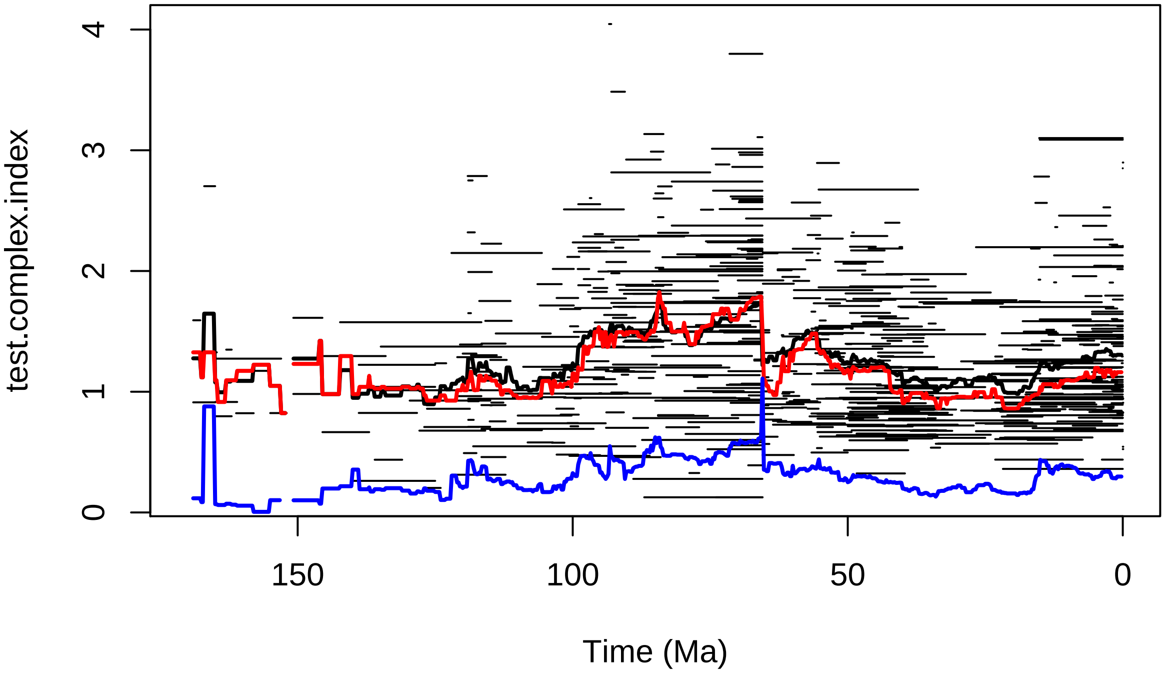
Yep, still got roughly the same expression. 2 survivor bugs are the ones at *almost* 0 down there. There's two if you look really really closely

```
morph$family->family.ID
range.line.plot(test.complex.index,
                "Test Complexity",
                family.ID,
                morph$origin,
                morph$extin
                )
```



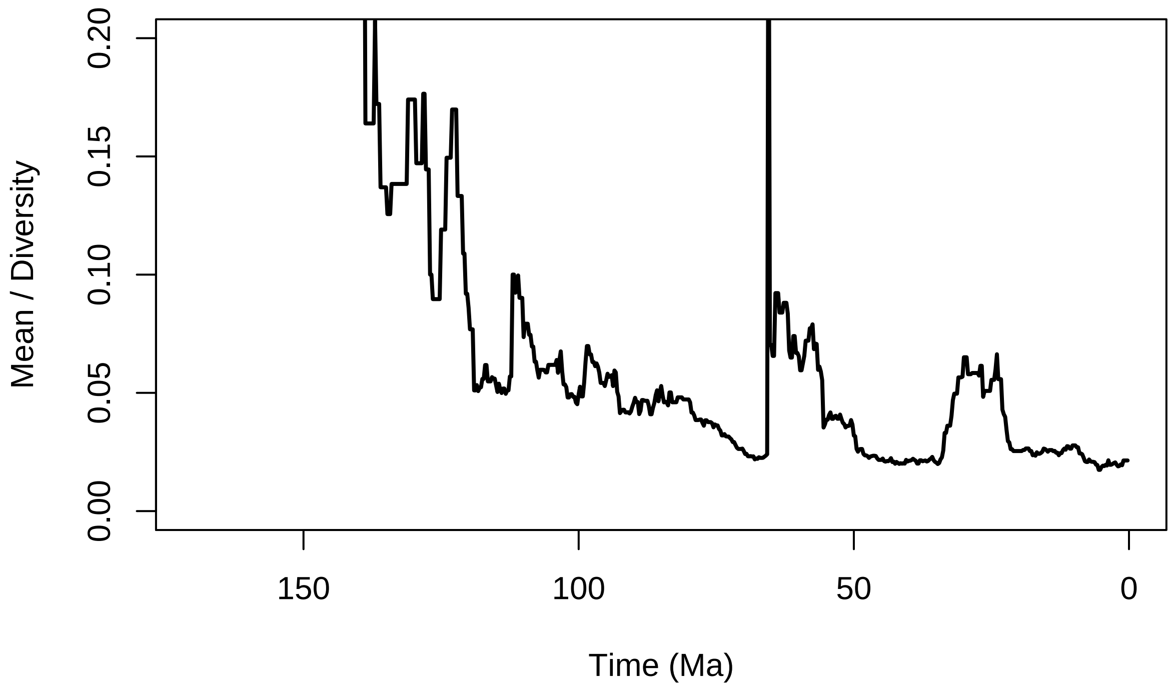
Here's the breakdown of the families.

```
plot(0,0,
     type='n',
     xlim=c(170,0),
     ylim=c(min(test.complex.index,na.rm=T),
            max(test.complex.index,na.rm=T)),
     xlab='Time (Ma)',
     ylab="test.complex.index")
segments(build.tci$origin,
         test.complex.index,
         build.tci$extin,
         test.complex.index)
lines(time.mean(test.complex.index,morph$origin,morph$extin,.25),lwd=2)
lines(time.median(test.complex.index,morph$origin,morph$extin,.25),lwd=2,col="red")
  time.var(test.complex.index,morph$origin,morph$extin,.25)->temp
  lines(temp[,1],temp[,2],lwd=2,col='blue')
```



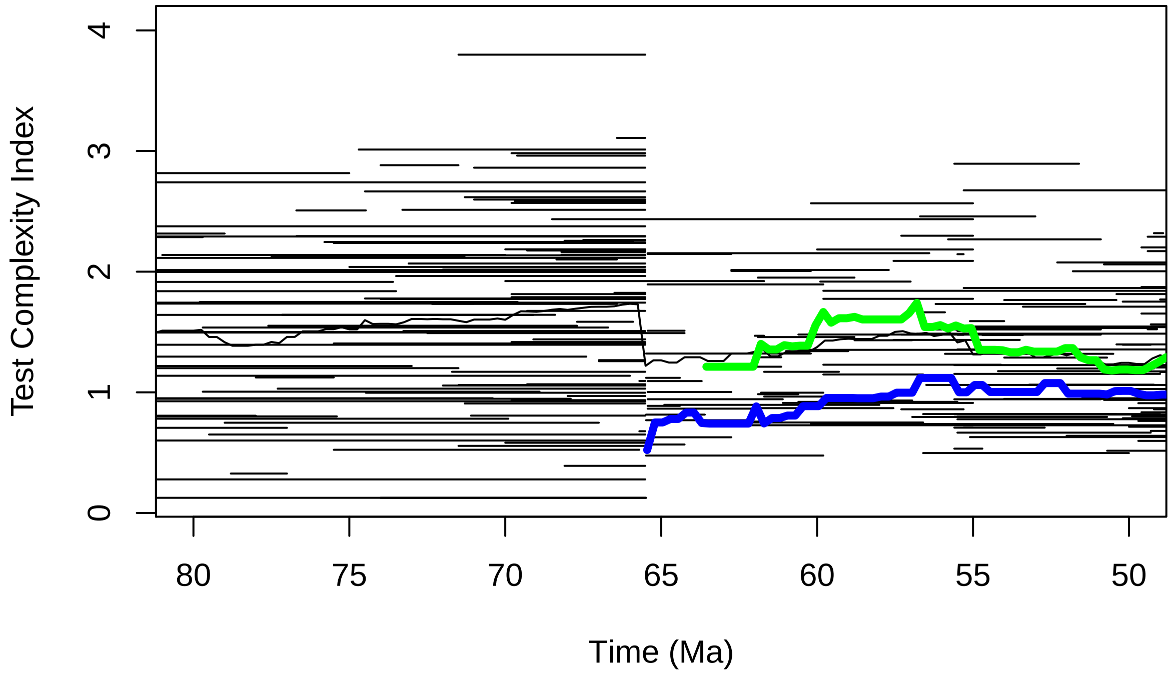
Alright, fine. Happy? The variance is mostly mimicking the mean. Yes, we can do the damn mean/diversity bit.

```
time.mean(test.complex.index,morph$origin,morph$extin,.25)[,2]/time.div(test.complex.index,morph$origin
plot(0,0,
     type='n',
     xlim=c(170,0),
     ylim=c(0,
            .2),
     xlab='Time (Ma)',
     ylab="Mean / Diversity")
lines(time.mean(test.complex.index,morph$origin,morph$extin,.25)[,1],meanperdiv,lwd=2)
```



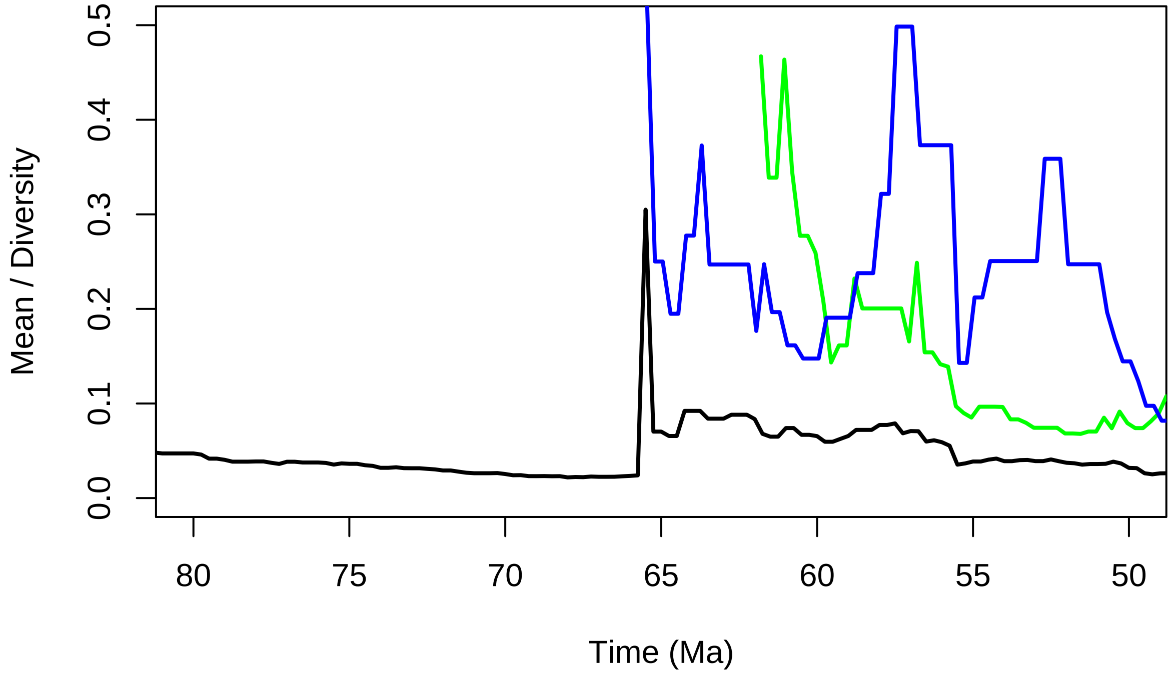
OK, back to the same story. Higher mean TCI scores during recovery as morphology is colonized (adaptive radiation).

```
photo.gen<-c('Morozovella',
             'Igorina',
             'Acarinina')
which(is.na(match(morph$genus,photo.gen))==FALSE)->photo.ind
c(photo.ind,which(morph$species == 'inconstans'))->photo.ind
#Spinose
spin.ind<-c("Globoturbotrotalia","Globorotalites","Catapsydrax","Paragloborotalia",
            "Subbotina", "Parasubbotina","Eoglobigerina")
which(is.na(match(morph$genus,spin.ind))==FALSE)->spin.ind
plot(0,0,
     type='n',

     #xlim=c(170,0),
     xlim=c(80,50),
     ylim=c(min(test.complex.index,na.rm=T),
            max(test.complex.index,na.rm=T)),
     xlab='Time (Ma)',
     ylab="Test Complexity Index")
segments(morph$origin,
         test.complex.index,
         morph$extin,
         test.complex.index)
#lines(time.mean(test.complex.index,morph$origin,morph$extin,0.25),lwd=2)
#[c(spin.ind,photo.ind)],

lines(time.mean(test.complex.index,morph$origin,morph$extin,0.25))
lines(time.mean(test.complex.index[photo.ind],
               morph$origin[photo.ind],
               morph$extin[photo.ind],0.25),
      lwd=4,
      col='green')
lines(time.mean(test.complex.index[spin.ind],
               morph$origin[spin.ind],
               morph$extin[spin.ind],0.25),
      lwd=4,
      col='blue')
```



Same
story

```
time.mean(test.complex.index,morph$origin,morph$extin,.25)[,2]/time.div(test.complex.index,morph$origin
time.mean(test.complex.index[photo.ind],morph$origin[photo.ind],morph$extin[photo.ind],.25)[,2]/time.di
time.mean(test.complex.index[spin.ind],morph$origin[spin.ind],morph$extin[spin.ind],.25)[,2]/time.div(te
plot(0,0,
     type='n',
     #xlim=c(170,0),
      xlim=c(80,50),
     ylim=c(0,
            .5),
     xlab='Time (Ma)',
     ylab="Mean / Diversity")
lines(time.mean(test.complex.index,morph$origin,morph$extin,.25)[,1],meanperdiv,lwd=2)
lines(time.mean(test.complex.index[photo.ind],morph$origin[photo.ind],morph$extin[photo.ind],.25)[,1],me
lines(time.mean(test.complex.index[spin.ind],morph$origin[spin.ind],morph$extin[spin.ind],.25)[,1],meanp
```


Mean / Diversity
0.0
0.1
0.2
0.3
0.4
0.5
80
75
70
65
60
55
50
Time (Ma)