Muusoctopus leioderma respiration

Lloyd Trueblood and Kirt Onthank

9/23/2021

Reading in libraries

```
library(OTools)
library(xlsx)
library(nlme)
library(car)
```

```
## Loading required package: carData
```

```
library(emmeans)
library(respirometry)
```

```
## Loading required package: PKNCA
```

```
library(knitr)
```

Find the relevant files

```
files=list.files(recursive=T)
resp.files=grep(".txt",files,value=T)
pcrit.files=grep("pcrit|pcrti",resp.files,value=T,ignore.case=T)
metab.files=setdiff(resp.files,pcrit.files)
blank.files=grep("blank_only",resp.files,value=T,ignore.case=T)
rmr.files=setdiff(metab.files,blank.files)
rmr.files=rmr.files[!grepl("-ch2.txt|-ch3.txt|-ch4.txt|\\(1\\).txt",rmr.files)]
rmr.files=rmr.files[!duplicated(basename(rmr.files))]
```

Reading in the data log xlsx file

```
data.log=read.csv("Muus_Data_Log.csv")
```

Running the RMR data analysis

First I am going to make a object to put the RMR data into

```
routine=data.frame(filename=as.character(),spreadsheet_guess=as.character(),octo=as.character(),mass=as
```

filename match check

```
file_check=as.character()
score=as.numeric()
for (i in 1:length(rmr.files)){
  filename=rmr.files[i]
  guess=which.min(adist(basename(filename),data.log$File.name))
  file_check[i]=data.log$File.name[guess]
  score[i]=min(adist(basename(filename),data.log$File.name))
}

write.csv(cbind(basename(rmr.files),file_check,score),file = "filecheck.csv")

#i=27
#cbind(t(adist(basename(rmr.files[i]),data.log$File.name)),basename(rmr.files),basename(rmr.files)[i])

#sum(is.na(resp$O23))<10&!grepl("blank",filename)
```

```
column.count=1
for (i in 1:length(rmr.files)){
  filename=rmr.files[i]
  print(paste("starting file ", basename(filename)," (loop",i,")",sep=""))
  if(length(grep("Group 4|presens|ch\\d\\.txt",basename(filename)))>0){
    resp=read.presens(filename)
  }else{
    resp=read.pyro(filename)
  }
  print("finding closest match in log")

guess=which.min(adist(basename(filename),data.log$File.name))

    flow=as.numeric(data.log$flow.rate..L.min.[guess])
    mass=as.numeric(data.log$Mass..g.[guess])
    if(is.na(flow)){
      flow=0.1
    }
    if(is.na(mass)){
      mass=10
    }
    print("calculating rmr")
    resp.mean=mean(resp.open(resp[resp$times>3600*3,],flow.rate=flow*1000,weight=mass)$resp,na.rm=T)
    print("writing data to object")
    routine[column.count,1]=basename(filename)
    routine[column.count,2]=data.log$File.name[guess]
    routine[column.count,3]=data.log$octo1[guess]
    routine[column.count,4]=mass
```

```
    if(length(grep("1800",filename))>0){
     routine[column.count,5]=1800
    }
    if(length(grep("1000",filename))>0){
      routine[column.count,5]=1000
    }
    routine[column.count,6]=data.log$day[guess]
    routine[column.count,7]=resp.mean
    column.count=column.count+1
    if(sum(is.na(resp$O23))<10&!grepl("blank",filename)){
      print("found second respirometer")
      flow=as.numeric(data.log$Flow.rate.2[guess])
      mass=as.numeric(data.log$Mass.2[guess])
      resp.mean=mean(resp.open(resp[resp$times>3600,],inflow=3,outflow=4,flow.rate=flow*1000,weight=mass
      print("writing data to object")
      routine[column.count,1]=basename(filename)
      routine[column.count,2]=data.log$File.name[guess]
      routine[column.count,3]=data.log$octo2[guess]
      routine[column.count,4]=mass

      if(length(grep("1800",filename))>0){
       routine[column.count,5]=1800
      }
      if(length(grep("1000",filename))>0){
        routine[column.count,5]=1000
      }
      routine[column.count,6]=data.log$day[guess]
      routine[column.count,7]=resp.mean
      column.count=column.count+1
    }
  print(paste("end of file ", basename(filename)," (loop",i,")",sep=""))
}
```

```
write.csv(routine,"RMR_Results.csv")
```

```
routine=read.csv("RMR_Results.csv")
```

Running linear effects model

log-log transformation

```
routine$mass.log=log(routine$mass)
routine$rmr.log=log(routine$rmr)
```

setting pCO2 to factor class:

```
routine$pco2=as.factor(routine$pco2)
```

Next I set orthogonal contrasts:

```
contrasts(routine$pco2)=contr.poly(2)
```

Running the linear mixed effects model and ANOVA using type III sum of squares:

```
routine.lme=lme(rmr.log~mass.log+pco2+day,random=~1|octo,
             correlation=corAR1(form=~day|octo),
              data=routine[routine$octo!="2-1",])
routine.anova=Anova(routine.lme,type="III")
routine.anova
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: rmr.log
##               Chisq Df Pr(>Chisq)
## (Intercept) 35.6056  1  2.416e-09 ***
## mass.log    13.0722  1  0.0002997 ***
## pco2         0.1150  1  0.7344687
## day          0.0572  1  0.8110012
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
rmr.lme.table=cbind(
  c("Mass","pCO2","Day"),
  round(routine.anova$Chisq[2:4],2),
  routine.anova$Df[2:4],
  round(routine.anova$`Pr(>Chisq)`[2:4],5)
)
colnames(rmr.lme.table)=c("Factor","Chi-square", "DF", "p-value")

kable(rmr.lme.table)
```

Factor	Chi-square	DF	p-value
Mass	13.07	1	3e-04
pCO2	0.12	1	0.73447
Day	0.06	1	0.811

```
predictions=data.frame(
  day=c(1,7,1,7),
  mass.log=log(rep(log(25.6),4)),
  pco2=as.factor(c(1000,1000,1800,1800))
)
```

```
exp(predict(routine.lme,newdata=predictions,level=0))
```

```
## [1] 5.630975 5.721444 6.108368 6.206507
## attr(,"label")
## [1] "Predicted values"
```

Regressions

```
#reg1.1800=nls(rmr~a*mass^b,data=routine[routine$pco2==1800&routine$day==1,],start=list(a=10,b=-0.7))
seq1.1800=seq(from=min(routine$mass.log[routine$pco2==1800]),
              to=max(routine$mass.log[routine$pco2==1800]),
              length.out=100)

df1.1800=data.frame(
  day=rep(1,100),
  mass.log=seq1.1800,
  pco2=as.factor(rep(1800,100))
)
pred1.1800= predict(routine.lme,newdata = df1.1800,level=0)

#reg1.1000=nls(rmr~a*mass^b,data=routine[routine$pco2==1000&routine$day==1,],start=list(a=10,b=-0.7))
seq1.1000=seq(from=min(routine$mass.log[routine$pco2==1000]),
              to=max(routine$mass.log[routine$pco2==1000]),
              length.out=100)

df1.1000=data.frame(
  day=rep(1,100),
  mass.log=seq1.1000,
  pco2=as.factor(rep(1000,100))
)

pred1.1000=predict(routine.lme,newdata = df1.1000,level=0)

#reg7.1800=nls(rmr~a*mass^b,data=routine[routine$pco2==1800&routine$day==7,],start=list(a=10,b=-0.7))
seq7.1800=seq(from=min(routine$mass.log[routine$pco2==1800]),
              to=max(routine$mass.log[routine$pco2==1800]),
              length.out=100)

df7.1800=data.frame(
  day=rep(7,100),
  mass.log=seq7.1800,
  pco2=as.factor(rep(1800,100))
)

pred7.1800=predict(routine.lme,newdata = df7.1800,level=0)

#reg7.1000=nls(rmr~a*mass^b,data=routine[routine$pco2==1000&routine$day==7,],start=list(a=10,b=-0.7))
seq7.1000=seq(from=min(routine$mass.log[routine$pco2==1000]),
              to=max(routine$mass.log[routine$pco2==1000]),
              length.out=100)
```

```
df7.1000=data.frame(
  day=rep(77,100),
  mass.log=seq7.1000,
  pco2=as.factor(rep(1000,100))
)

pred7.1000=predict(routine.lme,newdata = df7.1000,level=0)
```

```
plot(rmr~mass,data=routine[routine$octo!="2-1",],log="xy")
points(rmr~mass,data=routine[routine$pco2==1000&routine$day==1&routine$octo!="2-1",],pch=21,bg="white",
points(rmr~mass,data=routine[routine$pco2==1000&routine$day==7&routine$octo!="2-1",],pch=21,bg="blue")
points(rmr~mass,data=routine[routine$pco2==1800&routine$day==7&routine$octo!="2-1",],pch=21,bg="red")
points(rmr~mass,data=routine[routine$pco2==1800&routine$day==1&routine$octo!="2-1",],pch=21,bg="white",
lines(exp(seq1.1800),exp(pred1.1800),col="red",lwd=2,lty=2)
lines(exp(seq1.1000),exp(pred1.1000),col="blue",lwd=2,lty=2)
lines(exp(seq7.1800),exp(pred7.1800),col="red",lwd=2,lty=1)
lines(exp(seq7.1000),exp(pred7.1000),col="blue",lwd=2,lty=1)
```