Geometric morphometrics

Error analyses

Manuel F. G. Weinkauf

26–27 August 2022

Contents

1 Introduction **1**

2 Setting up the R session **2**

3 Error analysis **2**
3.1 For outline data 2
3.2 For landmark data 5

1 Introduction

Morphometric datasets are more prone to errors from various sources, that can significantly influence the results. Some of these errors are:

- **The device error**: The error introduced by the measuring device. This can take either of two forms:
  - *The error from the measurement device itself*: This occurs when you physically measure parameters, e.g. with a ruler or a vernier caliper. Each device has a maximum precision that can be reached, so measurements will never be perfect. This is more of an issue with traditional morphometrics, where normally lengths or weights are measured.
  - *The error from the image material*: In geometric morphometrics, you will often work with images to extract data. There, the cameras and scanners take images of a certain resolution, which limits the possible precision with which you can extract data. When you use microscopes or CT scanners in the process, you also introduce a limitation by the maximum achievable magnification and resulting optical resolution.
- **The definition error**: While landmarks should always be well-defined, some can be more precisely measured than others. If for instance a landmark is a nerve canal opening, this opening has a certain size (it is not a mathematical point), which introduces some ambiguity in the extraction process.
- **The error from material quality**: Especially with fossils, preservation is an issue. There may be deformations due to geological processes; some of them can be digitally removed but there remains some ambiguity.
- **The measurer error**: An error is introduced by the measurer in two differen ways:
  - *The error between measurers*: Different people will have slightly different perceptions of the position of a landmark, even if it is well-defined. As a result, different scientists will differ slightly in their measurements.
  - *The error betweens sessions*: Even the same person will generate slightly different measurements in different sessions. This is because (1) the environment while measuring (calm vs. loud) has an influence on measurement precision, (2) the daily form of the measurer (tired, exhausted, well-rested) is different across days and influences measurements, and (3) as measurer becomes more experienced with a set of specimens and the measurement template, their measurements will tend to become more precise.

As a result, methods were designed to quantify the measurement error in morphometrics based on an ANOVA approach. For more information see:

Yezerinac, S. M., Lougheed, S. C., and Handford, P. (1992) Measurement error and morphometric studies: Statistical power and observer experience. *Syst. Biol.* 41 (4): 471–82. doi: 10.1093/sysbio/41.4.471

2 Setting up the R session

For the error analysis, we will mainly use source code available in 'OutlineAnalysis_Functions.r' and 'GeometricMorphometrics_Functions.r'.

```
setwd("C:/R_Data/Erlangen_Morphometrics/Session6_ErrorAnalysis")
source("MorphoFiles_Function.r")
source("OutlineAnalysis_Functions.r")
source("GeometricMorphometrics_Functions.r")
```

3 Error analysis

3.1 For outline data

For outline data, the calculations are based on the harmonics of EFA analyses for two replications of data extraction. We can start by preparing our data:

```
#Read full dataset
Belemnite.Full<-Read.NTS("Belemnite_SmoothedOutline.nts")

#Split data into replications 1 and 2
Specimens<-unlist(dimnames(Belemnite.Full)[3])
Belemnite.R1<-Belemnite.Full[,,str_detect(Specimens, "R1")]
Belemnite.R2<-Belemnite.Full[,,str_detect(Specimens, "R2")]

#Calculate EFA harmonics
EFA.R1<-EFA.R2<-list()
Spec.Names<-strsplit(dimnames(Belemnite.R1)[[3]], split=".", fixed=TRUE)
for (i in 1:dim(Belemnite.R1)[3]) {
  EFA.R1[[i]]<-NEF(Belemnite.R1[,,i], Harmonics=15)
  names(EFA.R1)[i]<-Spec.Names[[i]][1]
}
Spec.Names<-strsplit(dimnames(Belemnite.R2)[[3]], split=".", fixed=TRUE)
for (i in 1:dim(Belemnite.R2)[3]) {
  EFA.R2[[i]]<-NEF(Belemnite.R2[,,i], Harmonics=15)
  names(EFA.R2)[i]<-Spec.Names[[i]][1]
}
```

EXERCISE 1: Given the equations you have available in the lecture, how would you go about calculating the error of our two EFA solutions?

For easier operation, we want to combine the harmonic coefficients of all replications into an array.

```
#Setting up array
Harm.Rep<-array(NA,
                dim=c(length(EFA.R1), length(EFA.R1[[1]][[1]])*4, 2),
                dimnames=list(names(EFA.R1),
                              c(paste(rep("A",
                                          length(EFA.R1[[1]][[1]])),
                                      1:length(EFA.R1[[1]][[1]]), sep="),
```

```
                                paste(rep("B",
                                          length(EFA.R1[[1]][[1]])),
                                      1:length(EFA.R1[[1]][[1]]), sep=""),
                                paste(rep("C",
                                          length(EFA.R1[[1]][[1]])),
                                      1:length(EFA.R1[[1]][[1]]), sep=""),
                                paste(rep("D",
                                          length(EFA.R1[[1]][[1]])),
                                      1:length(EFA.R1[[1]][[1]]), sep="")),
                              c("Rep1", "Rep2")))

#Replication 1
for (i in 1:length(EFA.R1)) {
  Temp<-EFA.R1[[i]]
  Harm.Rep[i,,"Rep1"]<-c(Temp[[1]], Temp[[2]], Temp[[3]], Temp[[4]])
}

#Replication 2
for (i in 1:length(EFA.R2)) {
  Temp<-EFA.R2[[i]]
  Harm.Rep[i,,"Rep2"]<-c(Temp[[1]], Temp[[2]], Temp[[3]], Temp[[4]])
}
```

With these two handy data frames, we can now calculate the average harmonic solution.

```
Harm.Mean<-apply(Harm.Rep, MARGIN=c(1, 2), FUN=mean)
```

Now that we have our data, we need to define the session factor and the individual factor in the replications. The session factor is easy, it is whether we work with replication 1 or 2. We can set up a factor-vector that encodes this through all rows of the array.

```
Session.factor<-gl(dim(Harm.Rep)[3], (dim(Harm.Rep)[1]))
```

The individual factor encodes the specimen. This means that we create another factor vector, where specimen 1 from replication 1 has the same factor as specimen 1 from replication 2; specimen 2 from replication 1 has the same factor as specimen 2 from replication 2; and so forth.

```
Individual.factor<-as.factor(rep((1:dim(Harm.Rep)[1]), dim(Harm.Rep)[3]))
```

We can now calculate the relative errors of the replications using a one-way ANOVA.

```
#Setting up results list
RelErr<-list()

for (i in 1:(dim(Harm.Rep)[2])) {
  Hm<-vector(length=0)
  #We combine data from both replications harmonic by harmonic
  for (j in 1:(dim(Harm.Rep)[3])) {
    Hm<-append(Hm, as.vector(t(Harm.Rep[,i,j])))
  }

  #We now calculate the session factor...
    SE<-summary(aov(Hm~Session.factor))
    {if (SE[[1]][2,3]>SE[[1]][1,3]) {pSE<-1} else {pSE<-0}}

    #...and the individual factor for this harmonic
```

```
    ME<-summary(aov(Hm~Individual.factor))
    {if (ME[[1]][1,3]>=ME[[1]][2,3]) {pME<-1} else {pSE<-0}}
    s2within<-MSwithin<-ME[[1]][2,3]
    MSamong<-ME[[1]][1,3]
    s2among<-(MSamong-MSwithin)/2
    Err<-s2within/(s2within+s2among)*100

    #And write everything in a neat report for inspection
    {if (i%%4==1) {
      RelErr$F1$Sessionfactor<-append(RelErr$F1$Sessionfactor,pSE)
      RelErr$F1$Individualfactor<-append(RelErr$F1$Individualfactor,pME)
      RelErr$F1$RelativeError<-append(RelErr$F1$RelativeError,Err)
     }
      else if (i%%4==2) {
          RelErr$F2$Sessionfactor<-append(RelErr$F2$Sessionfactor,pSE)
          RelErr$F2$Individualfactor<-append(RelErr$F2$Individualfactor,pME)
          RelErr$F2$RelativeError<-append(RelErr$F2$RelativeError,Err)
        }
        else if (i%%4==3) {
          RelErr$F3$Sessionfactor<-append(RelErr$F3$Sessionfactor,pSE)
          RelErr$F3$Individualfactor<-append(RelErr$F3$Individualfactor,pME)
          RelErr$F3$RelativeError<-append(RelErr$F3$RelativeError,Err)
          }
        else if (i%%4==0) {
          RelErr$F4$Sessionfactor<-append(RelErr$F4$Sessionfactor,pSE)
          RelErr$F4$Individualfactor<-append(RelErr$F4$Individualfactor,pME)
          RelErr$F4$RelativeError<-append(RelErr$F4$RelativeError,Err)
          }
    }
}
```

This output provides some interesting information. It lists per coefficient type (F1–F4) for each harmonic:

```
## $F1
## $F1$Sessionfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F1$Individualfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F1$RelativeError
##  [1] 0.057713645 0.009659959 0.057231668 0.074403177 0.019305457 0.075521608
##  [7] 0.131864403 0.066339554 0.008824088 0.041139548 0.159930241 0.189701826
## [13] 0.035170994 0.036763626 0.248585565
##
##
## $F2
## $F2$Sessionfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F2$Individualfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F2$RelativeError
```

```
##  [1] 0.0005973144 0.0294977557 0.0357742867 0.1687458191 0.0381068213
##  [6] 0.0338956590 0.1549545998 0.2704369288 0.0147257999 0.0708993696
## [11] 0.0991062263 0.0009164655 0.0325057510 0.0951989131 0.1506779863
##
##
## $F3
## $F3$Sessionfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F3$Individualfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F3$RelativeError
##  [1] 0.004194188 0.055825679 0.056265767 0.128217755 0.015051114 0.045035094
##  [7] 0.073129908 0.026905125 0.021917173 0.094000782 0.113273602 0.014283220
## [13] 0.069836229 0.076258273 0.344811501
##
##
## $F4
## $F4$Sessionfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F4$Individualfactor
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##
## $F4$RelativeError
##  [1] 0.023750301 0.044827560 0.144354904 0.047826740 0.053858707 0.148225627
##  [7] 0.251146530 0.002606077 0.027568109 0.033523114 0.100721979 0.036545178
## [13] 0.070187681 0.279867875 0.312376750
```

- If the session factor (replication) is significant. A significant result here would indicate that the two replications introduced an error that is larger than the variation between specimens, which would be problematic: 1—no systematic error during replication; 0—systematic error occurred, consider removing this component.
- If the individual factor is significant. An insignificant result here would indicate that the differences between specimens are less pronounced than the differences between replications in the same specimen, which would mean that observed differences between specimens are arbitrary: 1—variation between individuals larger than between replications; 0—variation between replications larger than between individuals, consider removing this component.
- The relative measurement error in per cent (fully comparable between studies).

The function 'OutlineAverage()' in 'OutlineAnalysis_Functions.r' provides the same results. At the moment, it requires harmonic coefficients being stored in .txt-files that are read automatically. In the future, I will possibly implement a version that can also be fed data directly as an R-object. The function was designed for EFA but should work for all other Fourier analysis types as well.

3.2 For landmark data

For landmark data, it is even less likely that we have a full replication of all data extractions, as this is so far done entirely manually and very time-consuming. For our *T. sacculifer*-data we have replications of tow samples, one with on average very small and one with on average very large specimens. They are available in the files '1439.5-1440_LM_Replicate1.tps', '1439.5-1440_LM_Replicate2.tps', '1488-1488.5_LM_Replicate1.tps', and '1488-1488.5_LM_Replicate2.tps'. We can start reading the data.

```
#Read data
S1.Rep1<-Read.TPS("1439.5-1440_LM_Replicate1.tps")
S1.Rep2<-Read.TPS("1439.5-1440_LM_Replicate2.tps")
S2.Rep1<-Read.TPS("1488-1488.5_LM_Replicate1.tps")
S2.Rep2<-Read.TPS("1488-1488.5_LM_Replicate2.tps")

#Combine data into list for further analysis
LM.Rep<-list(S1.Rep1$LMData, S1.Rep2$LMData, S2.Rep1$LMData, S2.Rep2$LMData)
```

EXERCISE 2: Given the equations you have available in the lecture, how would you go about calculating the error of our landmark replications?

We can now start the process of error calculation based on our two landmark extraction replications.

```
#Gather data dimensions
LM.Sam1<-list(LM.Rep[[1]], LM.Rep[[2]])
M<-length(LM.Sam1)#Number of datasets
N.1<-dim(S1.Rep1$LMData)[3]#Number of specimens, dataset 1
N.2<-dim(S2.Rep1$LMData)[3]#Number of specimens, dataset 2
LM<-dim(S1.Rep1$LMData)[1]#Number of landmarks
DM<-dim(S1.Rep1$LMData)[2]#Number of dimensions

#Combine data into vectors
Values<-vector(mode="numeric", length=0)
Session.factor<-vector(mode="numeric", length=0)
Individual.factor<-vector(mode="numeric", length=0)
for (i in 1:N.1) {
        Pos<-seq(from=i, to=M*N.1, by=N.1)
        Temp<-simplify2array(lapply(LM.Sam1, "[", , , i))
        Values<-append(Values, as.vector(Temp))
        Session.factor<-append(Session.factor, gl(dim(Temp)[3],
                                                  dim(Temp)[1]*dim(Temp)[2]))
        Individual.factor<-append(Individual.factor,
                                  rep.int(seq(from=((DM*LM)*(i-1)+1),
                                              to=((DM*LM)*(i-1)+1)+((DM*LM)-1),
                                              by=1), M))
    }
```

This process is a bit complicated. Essentially, because of the high dimensionality of the data, we take each specimen, pull out the data of both replicates of this specimen, write them into a vector, and append values for the session and individual factor on the fly.

We can now use these data to calculate the replication error.

```
#Convert session and indivisual factor into factors
Session.factor<-as.factor(Session.factor)
Individual.factor<-as.factor(Individual.factor)

#Calculate ANOVA
Session.Error<-summary(aov(Values~Session.factor))
Individual.Error<-summary(aov(Values~Individual.factor))

#Calculate error
s2within<-MSwithin<-Individual.Error[[1]][2,3]
MSamong<-Individual.Error[[1]][1,3]
s2among<-(MSamong-MSwithin)/M
```

```
Rel.Error<-(s2within/(s2within+s2among))*100
```

Again, we get some interesting output:

```
## [1] "Relative error"
```

```
## [1] 0.2642876
```

```
## [1] "Session error"
```

```
##                  Df  Sum Sq Mean Sq F value Pr(>F)
## Session.factor    1      52      52   0.013   0.91
## Residuals      2398 9843016    4105
```

```
## [1] "Individual error"
```

```
##                      Df  Sum Sq Mean Sq F value Pr(>F)
## Individual.factor 1199 9830050    8199   755.8 <2e-16 ***
## Residuals         1200   13018      11
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The relative error is relative small, so we did a good job here. We also see:

- The mean squares for the session error (52) is much smaller than for the residuals (4105) and the ANOVA is insignificant at $p = 0.91$. This shows us that the difference between specimens is much larger than between replications in the same specimen, so we do not need to assume a large measurement error.
- The mean squares for the individual error (8199) is much larger than for the residuals (11) and the ANOVA is significant at $p = 0$. This shows that te variation between specimens is much larger than any variation we introduce between replications by manually extracting the landmarks.

A function for this analysis is also available as 'LMAverage()' in 'GeometricMorphometrics_Functions.r'. This takes a character vector of the individual landmark datasets (as arrays in R) that should be averaged out and for which the errors should be calculated.

EXERCISE 3: Can you calculate the errors for the other sample for which we have a replication, to see if the error is larger in samples with smaller specimens?