Geometric morphometrics

Landmark analyses

Manuel F. G. Weinkauf

26–27 August 2022

Contents

1 Introduction

Landmark analyses are a bit more tricky to perform. As we loose degrees of freedom during the superimposition procedure, we have to modify the standard multivariate methods for this fact. This is why several packages offer modified versions of PCA, CVA, and other multivariate techniques for landmark analyses.

Here, we will go through some of these methods, using the our foraminifer dataset for continued work.

2 Setting up the R session

For landmark analyses, there are quite a few packages available with a large overlap between their functionalities.The package 'shapes' started in 2003. It has a tendency toward being more of a utility package, although it includes some analytical functions as well and defines a way to store landamrk data in an R-object that is pretty much standard to that day. The 'geomorph' package is a versatile package with very good documentation that saw the light of day in 2012. The package 'Morpho' is the youngest (first published in 2013) that tries to close some gaps in 'shapes' (the authors are partly the same) by offering a huge amount landmark analytical functions. The package 'Momocs' also contains some landmark analysis functionalities.

Beside, we will use some of my own code included in the 'MorphoFiles_Function.r' and 'GeometricMorphometrics_Functions.r' source files.

```
setwd("C:/R_Data/Erlangen_Morphometrics/Session5_LandmarkDataAnalysis")
library(shapes)
library(geomorph)
library(Morpho)
library(Momocs)
```

```
library(abind)
source("MorphoFiles_Function.r")
source("GeometricMorphometrics_Functions.r")
```

3 Reading and preparing the dataset

We will prepare the data according to the guidelines we worked out in the last exercise, using generalized Procrustes superimposition.

```
#Read data
Sacculifer<-Read.TPS("SacculiferLandmarks.tps")

#Superimpose landmarks
Sacculifer.GPA<-procGPA(Sacculifer$LMData)
```

```
## [1] "To speed up use option distances=FALSE"
## [1] "To speed up use option pcaoutput=FALSE"
```

We also have covariates for the foraminiferal data. The file 'ForamAbundances.txt' contains abundance data of *T. sacculifer* and another species of planktonic Foraminifera (*Orbulina universa*) throughout the core section. The file 'ForamOceanography.txt' contains date for sample age, $\delta^{18}O$-values of the planktonic foraminifer *Globigerinoides ruber albus* (indicating temperature) and the reconstructed relative sea level (directly inversely correlated to salinity in the Red Sea).

```
Foraminifera.Abundances<-read.table("ForamAbundances.txt", header=TRUE, sep="\t")
colnames(Foraminifera.Abundances)<-c("Depth.in.core.cm", "Age.ka",
                                     "O.universa.accumRate", "O.universa.relAbund",
                                     "T.sacculifer.accumRate", "T.sacculifer.relAbund")

Foraminifera.Oceanography<-read.table("ForamOceanography.txt", header=TRUE, sep="\t")
colnames(Foraminifera.Oceanography)<-c("Depth.in.core.cm", "Age.ka", "d18O.ruber", "RSL")
```

4 Exploratory data analysis

As in outline analyses, it is a good idea to start with some exploratory data analyses to learn to know the data better and find potential groupings in the data.

4.1 Principal component analysis

Principal component analysis is as valuable in landmark analyses as it is in outline analysis. However, we cannot use the standard PCA here. Due to the fact that the superimposed landmarks are collinear (i.e. if you move one all others have to move as well to preserve the shape) we are loosing degrees of freedom in the analysis: $-4df$ in the 2D-case and $-7df$ in the 3D case.

One function that takes these circumstances into account is the 'geomorph'-function 'gm.prcomp()' (Fig. 1):

```
#Calculate PCA
PCA1<-gm.prcomp(Sacculifer.GPA$rotated)

#Calculate proportion of variance explained per axis
PoV<-(PCA1$sdev^2/sum(PCA1$sdev^2))*100

#Plot results
plot(PCA1$x[,1], PCA1$x[,2], type="p",
     xlab=paste("PC 1 (", round(PoV[1], digits=0), "%)", sep=""),
```

```
     ylab=paste("PC 2 (", round(PoV[2], digits=0), "%)", sep=""), pch=16,
     cex=0.5, col="black")
```

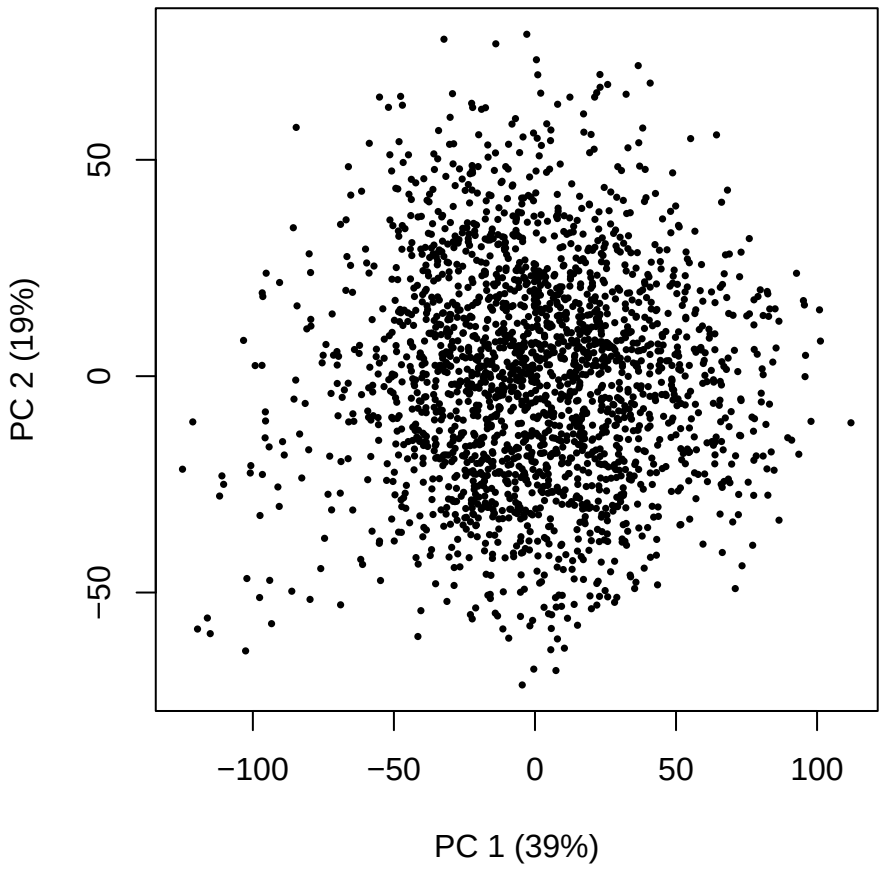

Figure 1: PCA of Red Sea Foraminifera.

Another, more complex version of this procedure is available as 'LM.PCA()' in 'GeometricMorphometrics_Functions.r'. It plots output directly (currently), and offers the three PCA types (Fig. 2):

- *Explorative*: Offers an overview of the PCA morphospace and the shapes at its extreme endpoints.
- *Uniform*: Displays the uniform deformation across the morphospace.
- *Nonaffine*: Displays relative warps across the morphospace.

```
#Calculate explorative PCA
PCA2<-LM.PCA(Sacculifer.GPA$rotated, Type="Explorative",
             Lines=c(1, 2, 3, 4, 5, NA, 10, 9, 8, 7, 6, 11, 12))
```

```
## [1] "To speed up use option distances=FALSE"
## [1] "To speed up use option pcaoutput=FALSE"
```

EXERCISE 1: Explore the capabilities of the landmark PCA. Which groupings could you think of in the data to explore?

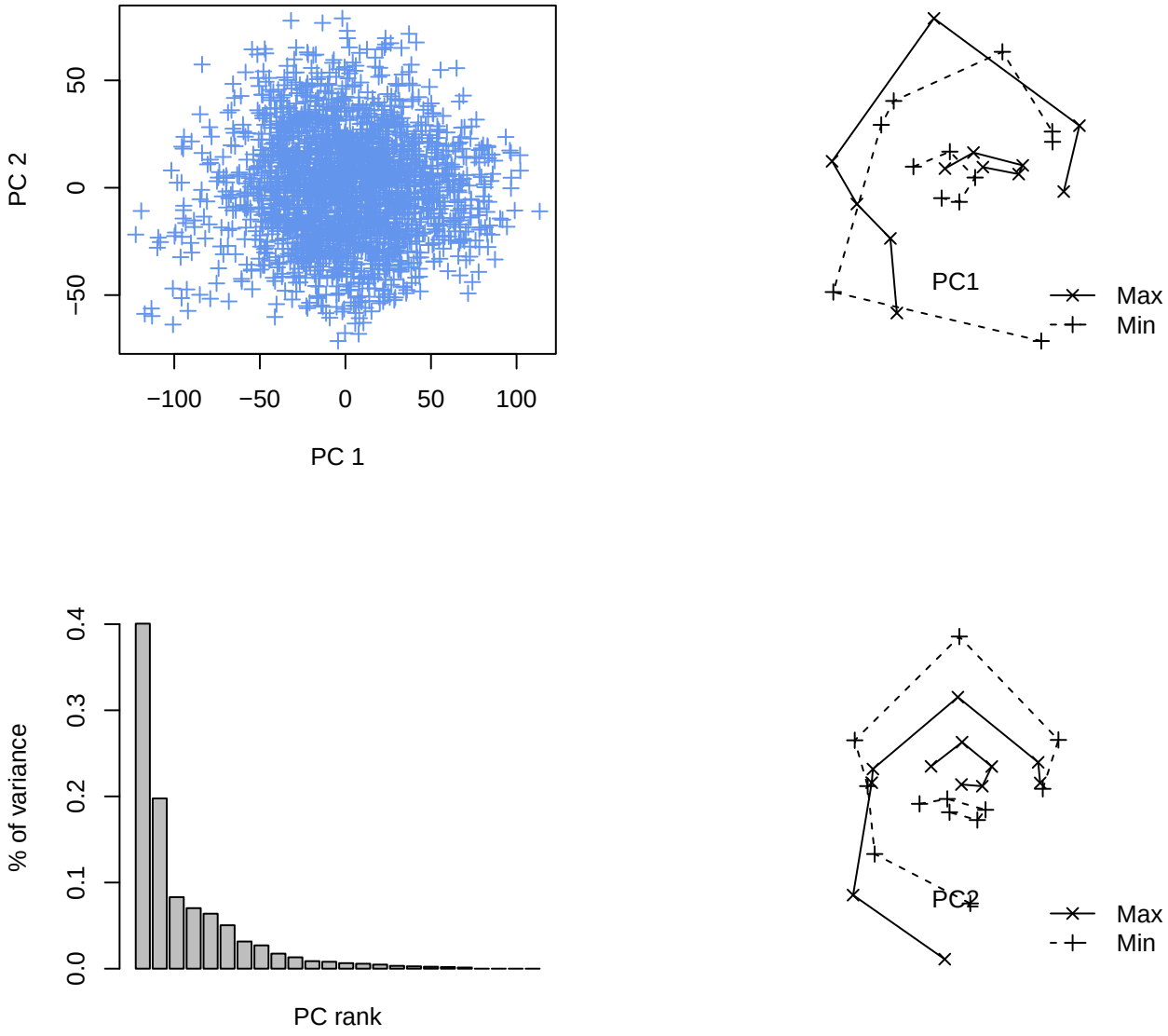

Figure 2: Explorative PCA of Red Sea Foraminifera.

5 Hypothesis-driven data analysis

5.1 Morphological integration

Normally, you would have certain hypothesis that drive your project. One potential question could be, whether parts of the shape of your organisms are morphologically integrated. Morphological integration happens, when a certain subset of landmarks is not able vary independently of another. We then say, they are morphologically integrated. Such structures cannot be considered individually for any adaptive or evolutionary processes, as a change in one part necessitates a change in another part.

The 'geomorph'-package offers the 'integration.test()'-function to evaluate this. It takes the Procrustes shapes coordinates, and a vector indicating which landmarks belong to which potential module.

We could for instance ask, whether the foramen can vary in shape independently from the rest of the shell. The template in Exercise 4 shows us, that the foramen is described by the first five landmarks, the remaining seven landmarks belong to the shell (Fig. 3).

```
#Calculate morphological integration
modules<-c(rep("A", 5), rep("B", 7))
Int.Test<-integration.test(Sacculifer.GPA$rotated,  partition.gp=modules)
```

```
##
## Random PLS calculations: 1000 permutations.
##   |                                                                      |
```

```
summary(Int.Test)
```

```
##
## Call:
## integration.test(A = Sacculifer.GPA$rotated, partition.gp = modules)
##
##
##
## r-PLS: 0.816
##
## Effect Size (Z): 12.9085
##
## P-value: 0.001
##
## Based on 1000 random permutations
```

```
#Plot results
plot(Int.Test)
```

In this example, we see a strong and significant relationship, which is rather expected. Such analyses are performed via partial-least-squares regression (PLS), that allow a regression analysis to be conducted between two multivariate datasets. The general 'geomorph'-function for that, where you can for instance test the relationship between morphology and another multivariate and collinear dataset, is calles 'two.b.pls()'.

EXERCISE 2: Test for other feasible morphological integrations or other more general PLS scenarios.

5.2 Analysing group differences

One of the major application of geometric morphometrics is to test, whether pre-defined groups differ in morphology and if so, how. We can start by dividing the data into two sets based on abundance (Fig. 4).

```
#Extract core-depth information from samples
Depths<-strsplit(Sacculifer$ID, split="_", fixed=TRUE)
Depths<-unlist(lapply(Depths, "[[", 1))
```

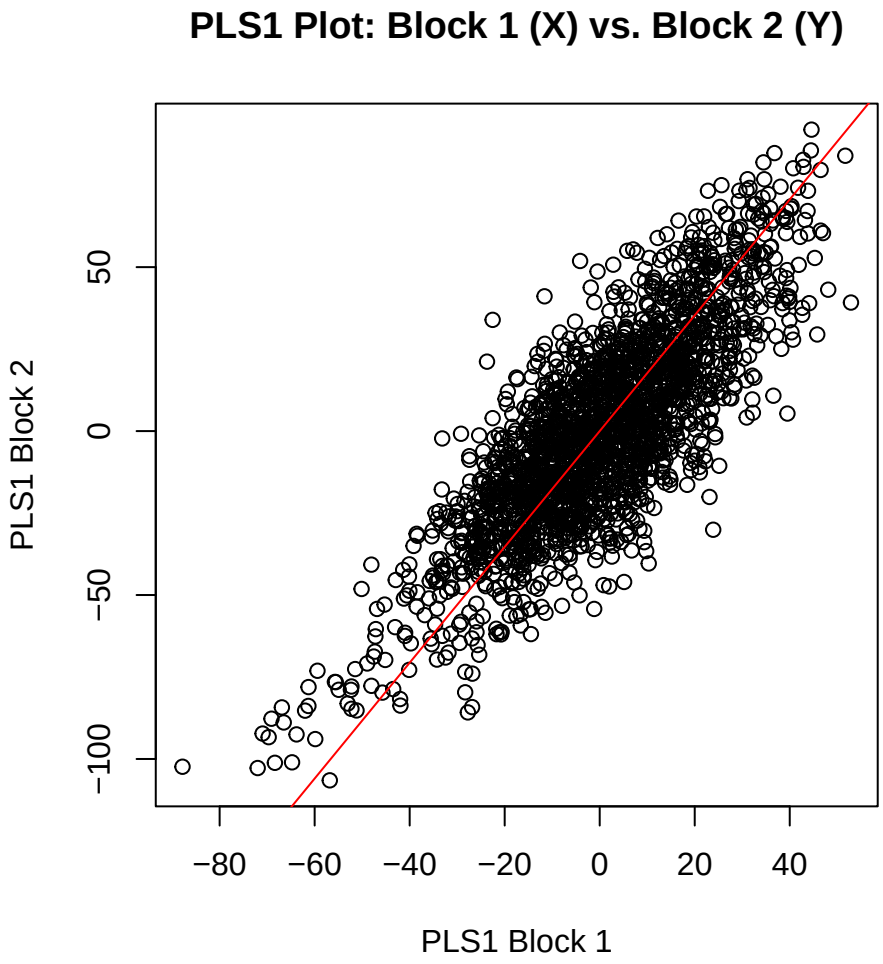

Figure 3: Testing foraminiferal shapes for morphological integration.

```
Depths<-strsplit(Depths, split="-", fixed=TRUE)
Depths<-lapply(Depths, as.numeric)
Depths<-unlist(lapply(Depths, mean))

#Select abundance level for separation
hist(Foraminifera.Abundances[,"T.sacculifer.accumRate"], col="grey", main="",
     xlab="Abundance")
```

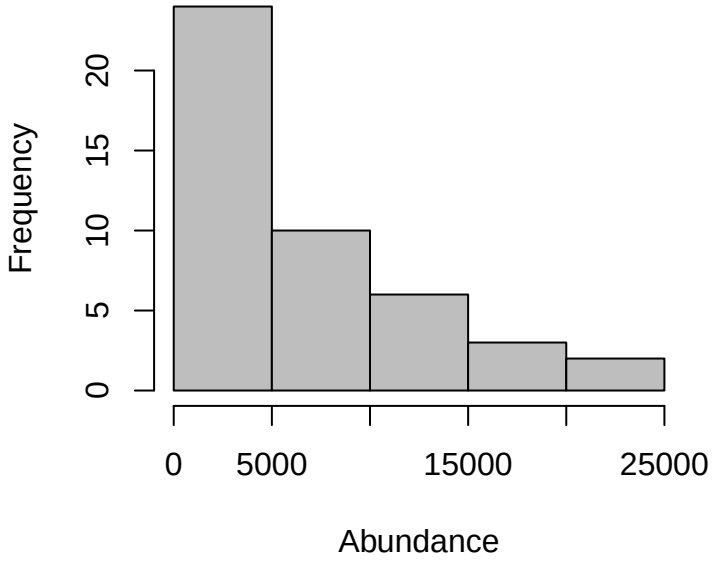

Figure 4: Histogram of *T. sacculifer* abundances.

Let's use 10000 as a threshold value to divide a low-abundance and a high-abundance population.

```
#Find sample depths with abundances below and above the threshold
Low<-Foraminifera.Abundances[which(Foraminifera.Abundances[,"T.sacculifer.accumRate"]
                                   <=10000),"Depth.in.core.cm"]
High<-Foraminifera.Abundances[which(Foraminifera.Abundances[,"T.sacculifer.accumRate"]
                                    >10000),"Depth.in.core.cm"]

#Divide landmark data into two sets
Low.Pos<-which(Depths %in% Low)
Sac.Low<-Sacculifer.GPA$rotated[,,Low.Pos]
High.Pos<-which(Depths %in% High)
High.Pos<-High.Pos[which(High.Pos<=dim(Sacculifer$LMData)[3])]
Sac.High<-Sacculifer.GPA$rotated[,,High.Pos]
```

With these data, we can now check for morphological differences caused by ecological stress.

5.2.1 Using canonical variates analysis

We can use the canonical variates analysis (CVA) you alreadyay know to test for group differences between the two groups. An implementation for landmark data is availabe as 'CVA()' in 'Morpho' (Fig. 5).

```
#Prepare data
CVA.Data<-abind(Sac.Low, Sac.High, along=3)
Abund<-c(rep("Low", dim(Sac.Low)[3]), rep("High", dim(Sac.High)[3]))

#Calculate CVA
```

```
Sac.CVA<-CVA(CVA.Data, as.factor(Abund), p.adjust.method="BY")
```

```
## singular Covariance matrix: General inverse is used. Threshold for zero eigenvalue is 1e-10
```

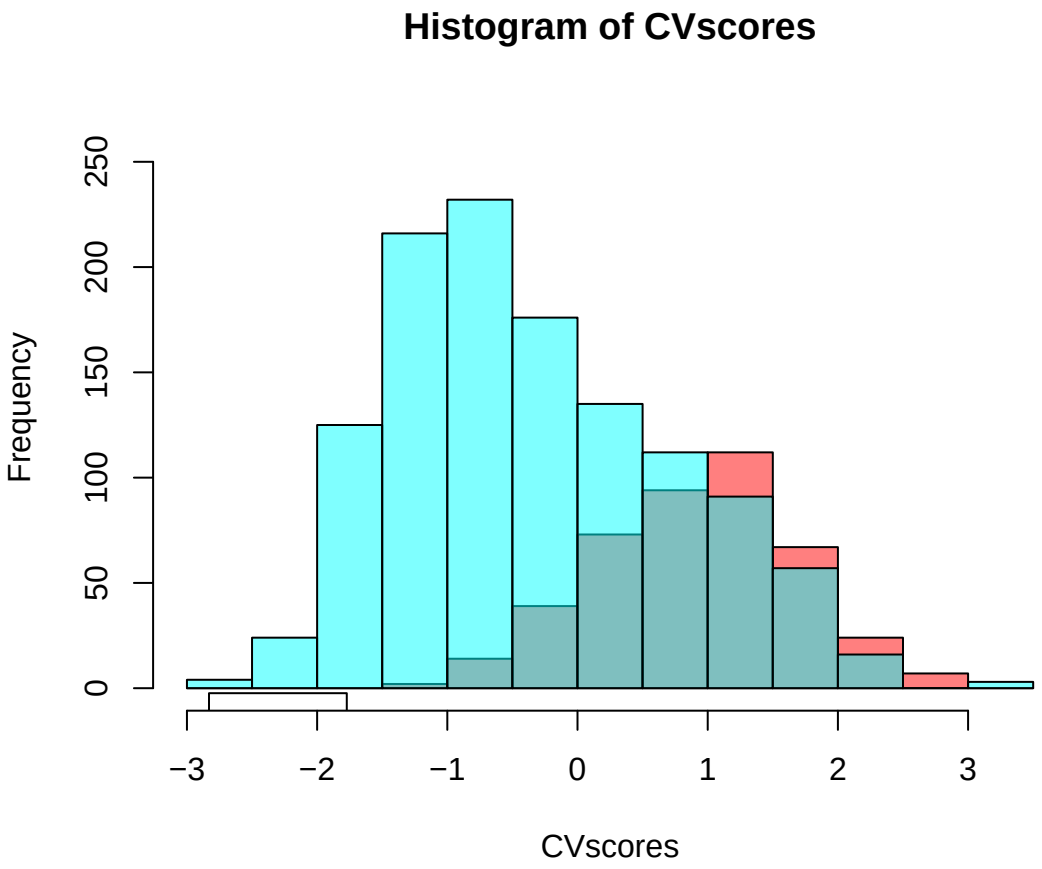

Figure 5: CVA of morphology between low- and high-abundance groups.

```
Sac.CVA
```

```
##  classification result in frequencies
##
##        High  Low
##   High  175  257
##   Low   140 1051
##
##
##  classification result in %
##
##          High    Low
##   High 40.509 59.491
##   Low  11.755 88.245
##
##
##  overall classification accuracy: 75.53913 %
```

5.2.2 Displaying group differences

We can also show how the shape changes between these two groups, for instance using the 'tps()'-function in 'GeometricMorphometrics_Functions.r' (Fig. 6).

```
tps(Sacculifer.GPA$rotated, Low.Pos, High.Pos, n=20, Ampl=FALSE,
    Lines=c(1, 2, 3, 4, 5, NA, 10, 9, 8, 7, 6, 11, 12), Points=c(3, 4),
    Col=c("red", "blue"), Scale=1, Line.width=2,
    Legend.Text=c("Low abundance", "High abundance"))
```

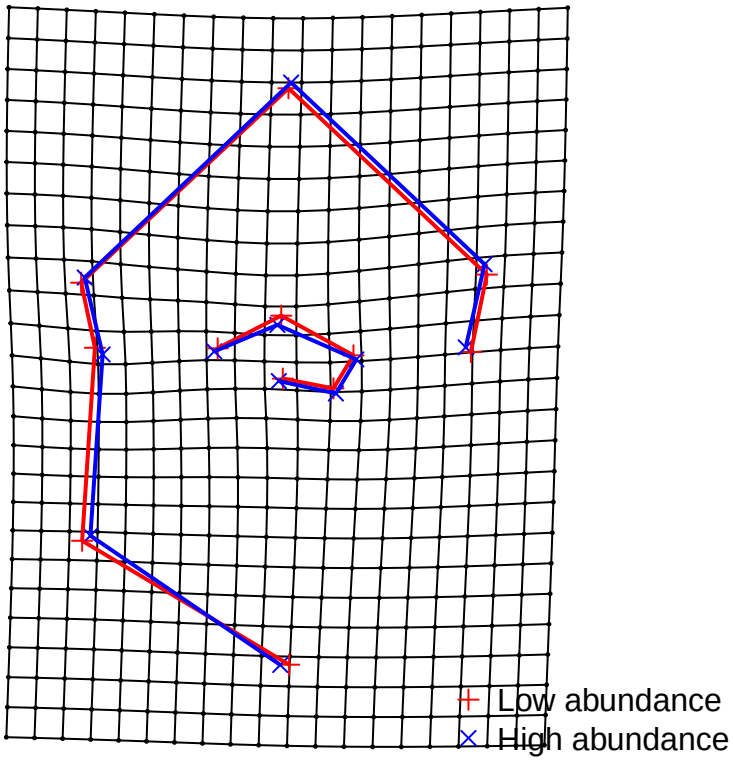

Figure 6: Thin-plate spline deformation between low and high abundance group.

EXERCISE 3: Explore the functions to test for morphological group differences some more.