

Procrustes analysis



Euclid

Today's Session

9:30 AM - 10:30 am Details about Procrustes

10:30-11:00 am Coffee Break

11:00 am - 12:30 pm Details about PCA, Eigenvectors, Eigenvalues

12:30 - 2:00 pm Lunch Break

2:00 - 3:00 pm Statistics in R

3:00 - 4:30 pm Tarsal exercise

4:00 - 5:00 pm Wrap up

digitize2d()

Erik Otarola-Castillo has sent a fix. Do the following and the function will work:

- 1. Set the working directory to where your images are stored.
- 2. Define the function picscale() as follows

```
picscale <- function(scale){
   digscale<-NULL
   digscale<-locator(2,type="o",lwd=2,col="red",lty="11")
   sqrt(sum(diff(digscale$x)^2 + diff(digscale$y)^2))*scale
}</pre>
```

To collection landmarks in digitize2d:

- 1. If you have a scale bar in the images
 - 1.1. enter the length of the scale bar as argument (e.g., 10 if the scale bar is 10 mm)
 - 1.2. when digitize2d starts, click on both ends of the scale bar, then click on your landmarks in the proper order
 - 1.3. coordinates and centroid sizes will be scaled in the units of your scale bar (e.g., mm)

2. If you don't have a scale bar:

- 2.1. enter 1 as the scale argument
- 2.2. when digitize2d starts, click in two arbitrary but different places on the image, then click on your landmarks in the proper order
- 2.3. coordinates will have no real units and centroid size will be meaningless

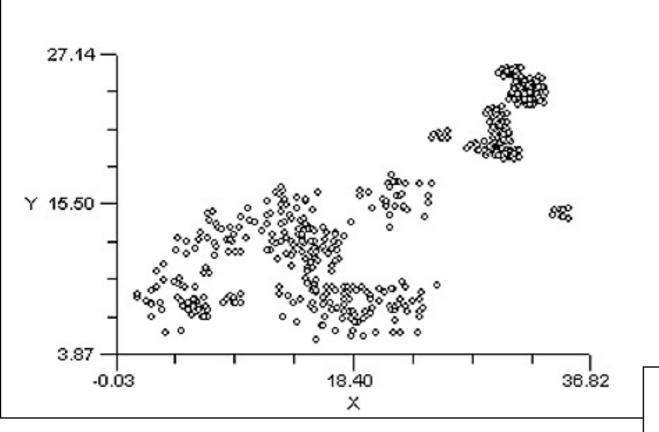
Procrustes superimposition

also known as...

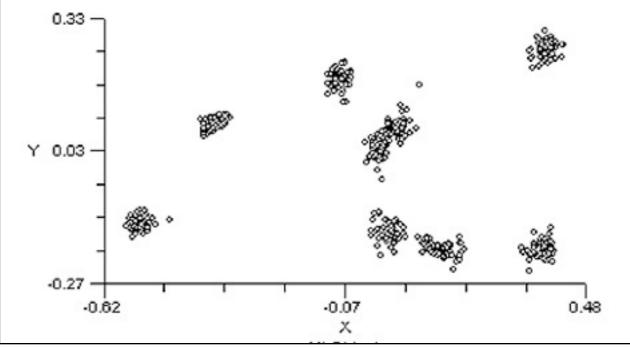
Procrustes analysis
Procrustes fitting
Generalized Procrustes Analysis (GPA)
Generalized least squares (GLS)
Least squares fitting

- Centers all shapes at the origin (0,0,0)
- Usually scales all shapes to the same size (usually "unit size" or size = 1.0)
- Rotates each shape around the origin until the sum of squared distances among them is minimized (similar to least-squares fit of a regression line)
- Ensures that the differences in shape are minimized

Procrustes superimposition





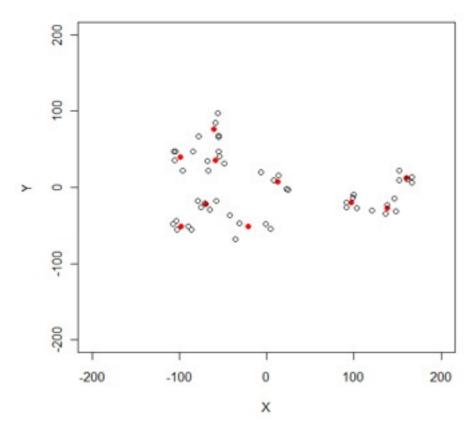


After Procrustes

Definitions

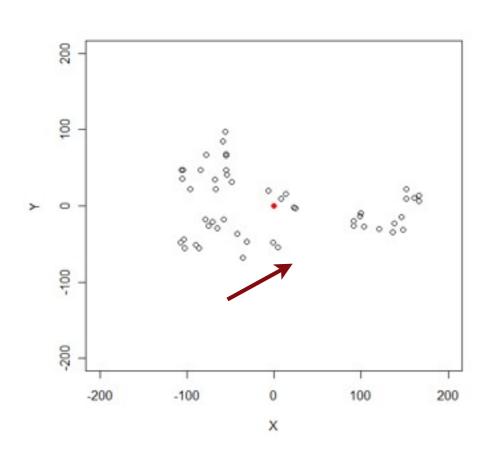
Shape - a configuration of landmarks irrespective of size ("form" is sometimes used to distinguish a set of landmarks that have a scale)

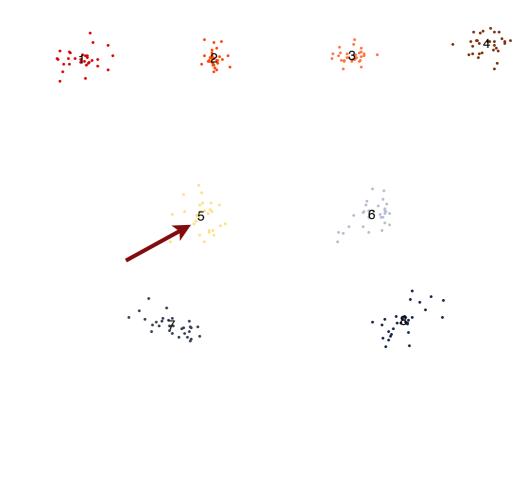
Consensus shape - the mean shape of a sample



More definitions

Centroid - the center point of a shape, a sample of shapes, or a single landmark in a sample of shapes

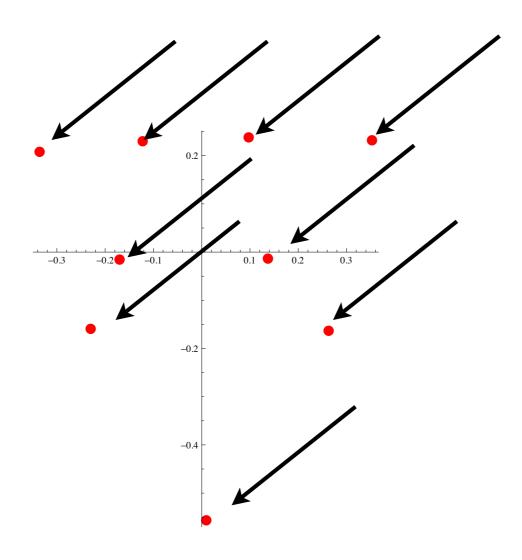




Procrustes: Translation, Rotation, Rescaling

Thought problem...

How do you translate one of your original landmark shapes to the point {0,0} on a graph?



Goal of morphometrics: measuring similarity and difference

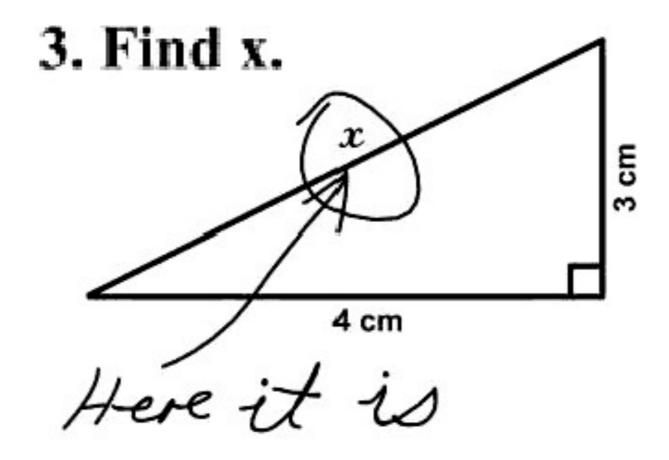
Morphometric distances are the main measure of difference

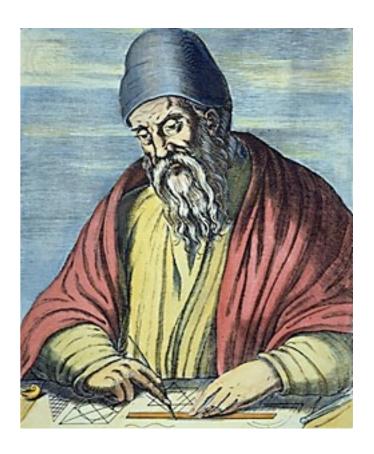
Measured as the difference between objects (which might be specimens or means of species, or whatever) on all the variables being used

In geometric morphometrics, the main measure of difference is the Procrustes distance, the distance between shapes after they have been superimposed

Morphometric distances = Euclidean distances (more or less)

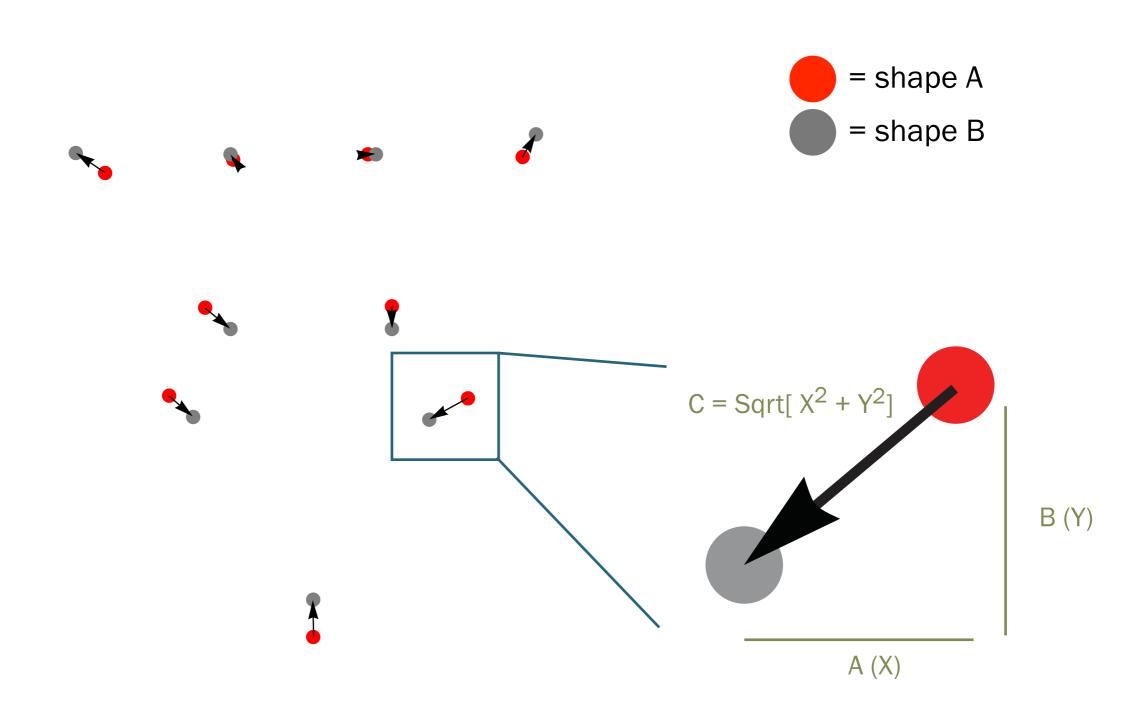
Euclidean distance = $sqrt(a^2 + b^2)$





Procrustes distance

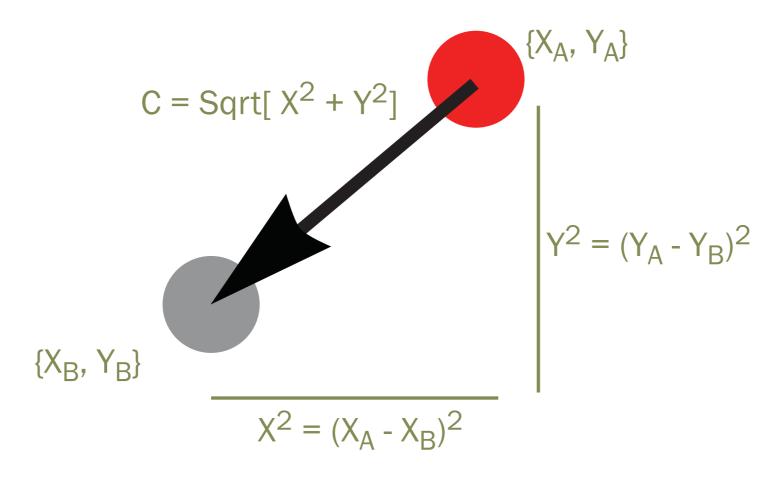
Sum of the distances between corresponding landmarks of two shapes.



Procrustes distance in Mathematica

How would you calculate the Procrustes distance between the mean shape (consensus) and any one of the objects?

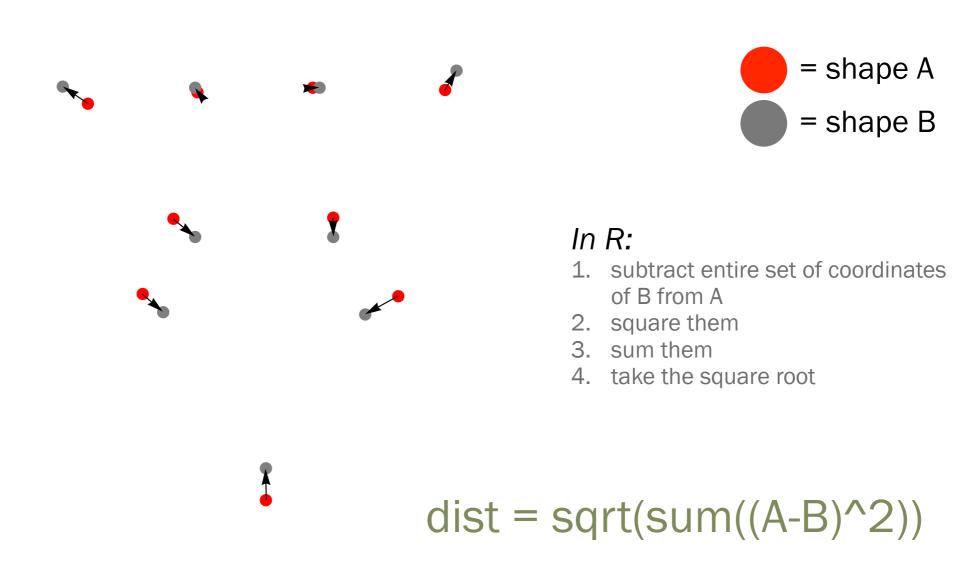
1. How to calculate X^2 ? Y^2 ?



$$({X_A, Y_A} - {X_B, Y_B}) ^2$$

Procrustes distance

Sum of the distances between corresponding landmarks of two shapes.



Use Procrustes distance to find outliers

- Q. Which faces are most different from the average?
- A. The ones most distant from the mean shape (consensus).

In R:

- 1. consensus = apply(proc\$coords, c(1,2), mean)
- 2. dists <- array(dim=dim(proc\$coords)[3])
- 3. for(i in 1:dim(proc\$coords)[3]) { dists[i] <- sqrt(sum((proc\$coords[i]-consensus)^2)) }
- 4. sort(dists)



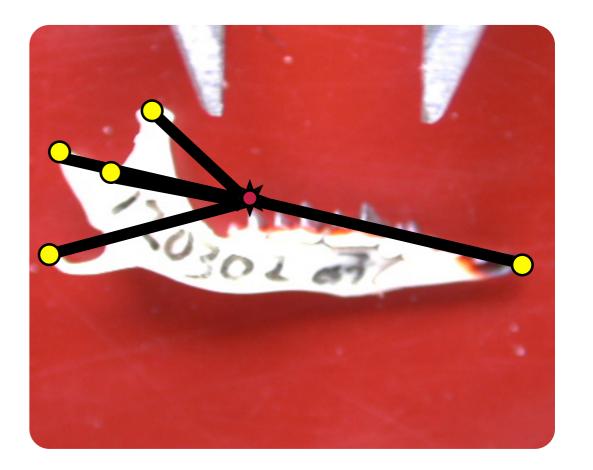
Centroid Size

standard measure of "size" of a shape in geometric morphometrics

Centroid size—sum of the distances from the centroid to each individual landmark

Size is removed from specimens by scaling each one so that centroid size = 1.0.

Calculated from <u>original landmark</u> <u>coordinates</u> (not Procrustes coordinates) and <u>only if they have been scaled</u> to real units (e.g., mm)



Calculate centroid size in R

```
> shape <- proc$coords[,,1]
> centroid <- apply(shape,2,mean)
> centroidsize <- sqrt(sum((t(t(shape)-centroid))^2))</pre>
```

Check to see what the centroid and centroid size are. Why do they have these values?

Translation, scaling, and rotation

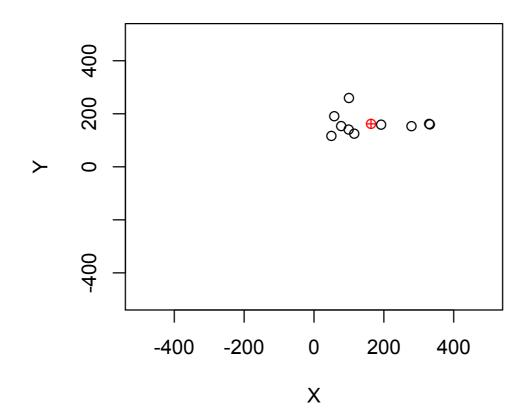
the three components of Procrustes

The next slides take us through the steps required to center (translate), resize, and rotate a set of landmarks. These changes are called "rigid transformations" because they do not change the relative positions of the landmarks. Try each of these steps for yourself.

Step 1

Plot the original landmarks of a shape on a large empty graph:

- > shape <- lands[,,1]
- > centroid <- apply(shape,2,mean)</pre>
- > centroidsize <- sqrt(sum((t(t(shape)-centroid))^2))</pre>
- > plot(c(-500,500),c(-500,500),type="n",xlab="X",ylab="Y")
- > points(shape)
- > points(t(centroid), pch=19,col="Red")



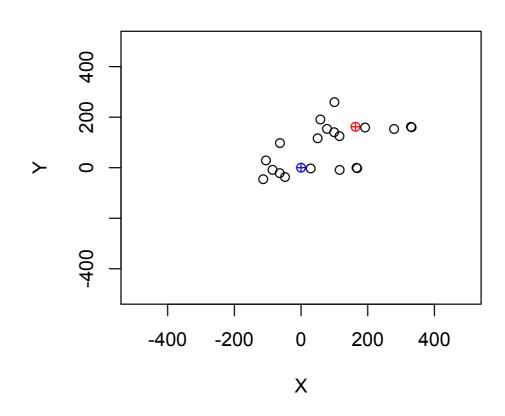
Step 2: translate

Subtract the centroid of the shape from each pair of landmark coordinates.

Because colMeans() was used to calculate the centroid, it is in dataframe format and you have to transform it into a matrix of the right shape and layout to subtract it from the shape. Plot the new centered shape and its new centroid on the same graph. This is an example of matrix addition or subtraction.

- > newshape <- t(t(shape) centroid)
- > points(newshape)
- > newcent <- apply(newshape,2,mean)
- > points(t(newcent),pch=10,col="Blue")

What kind of transformation have you just done? What are the coordinates of the new centroid?



Step 3: scale

Calculate the centroid size of the original shape and the new shape.

As in 2, the centroid has to be reformatted into a matrix to subtract it from the shape.

> sqrt(sum((t(t(newshape)-centroid))^2))

How has the centroid size changed?

Can you see a shortcut to calculating the centroid size of the centered shape?

Step 4

Divide the landmark coordinates of the centered shape by their centroid size and plot them on the same graph. This is an example of multiplying or dividing a matrix by a scalar.

> resizedshape <- newshape / centroidsize

Why can't you see the points on the old graph?

What is the centroid size of the resized shape?

Create a new plot so you can see the resized points.

How could you transform the shape so that its centroid size equals 2.0?

Angles in Geometric Morphometrics

Angles (in radians) are used for rotating data, as in Procrustes

- Angles are used as measures of distance in shape space
- Eigenvector values are angles of rotation (in radians)
- Angles are often symbolized with the Greek letter Theta (θ)

Radians

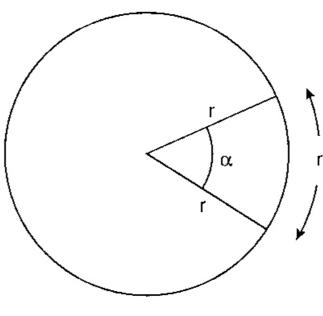
Radians are units of angular measurement

1 radian equals the angle required to trace an arc whose length equal to the radius of the circle.

There are 2 pi radians in a complete circle (360°)

$$1 \text{ rad} = 180 / \text{pi} = 57.3^{\circ}$$

Radians are useful because they are mathematically easier to work with than degrees, minutes and seconds



 $\alpha = 1$ radian

Step 5: rotate

Create a rotation matrix to rotate the landmarks by first calculating the angle in radians, then plugging it into the rotation matrix formula for x,y points, namely,

$$\begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

Plot the landmarks in their original position, then rotated at 45° intervals through a full circle. This is an example of one of several kinds of matrix multiplication, whose symbol in R is "%*%" rather than "*".

```
> angle <- 45 * pi / 180
```

- > rotmat <- matrix(c(cos(angle),-sin(angle),sin(angle),cos(angle)),byrow=T,ncol=2)
- > plot(c(-1,1),c(-1,1),xlab="x",ylab="y", type="n")
- > points(resizedshape)
- > points(resizedshape%*%rotmat, pch=19,col="Red")

Can you write a function that rotates the landmarks at any angle you specify?

Procrustes removes degrees of freedom

Degrees of freedom are lost from shape data because of Procrustes

- 1 degree of freedom lost for translation in each dimension (2 for 2D, 3 for 3D)
- 1 degree of freedom lost of scaling
- 1 degree of freedom lost for rotation in 2D, 3 degrees lost for rotation in 3D

Total: 4 degrees lost in 2D and 7 degrees lost in 3D

Thus, the last 4 PCs have zero variance because that variance is lost to Procrustes.

Summary

Procrustes in technical terms...

Procrustes translates, rescales, and rotates landmarks into optimal superimposition.

Translation puts the centroid of all shapes at [0,0]

Rescaling gives all shapes a centroid size = 1.0

Rotation iteratively rotates them so that the Procrustes distances between them is minimized (equal to minimizing the sum of squared distances between them)

Reasons for Procrustes...

Removes the effects of size so that the first principal component is pure shape Makes landmark configurations comparable by putting them in the same coordinate system

Promotes hypothesis testing by minimizing the distances, ensuring that any difference between shapes is a true difference in their "shape"