

Contents

1 Computing the distance between two shapes

1

1 Computing the distance between two shapes

In the previous exercise, you learned how center and scale 2D shapes in landmark shape space and the continuous curve shape space. That was just two similarity transformations which are commonly ascribed to the shape of an object. Rotation and often reflection are also commonly considered to be shape invariant (or similarity) transformations.

Let $C \in \mathbb{R}^{n \times 2}$ denote the shape matrix. Let $C_{i,j} \in \mathbb{R}$ denote the elements of the shape for $i \in \{0, \dots, n-1\}$ and $j \in \{0, 1\}$. Let $C_i = (C_{i,0}, C_{i,1})^T \in \mathbb{R}^2$ denote the 2D coordinates of the shape.

The landmark based shape space can be thought of as an orbit of points in $\mathbb{R}^{m \times 2}$ over all similarity transformations.

$$[C] = \{(sC + \mu)R \mid R \in O(2), s > 0, \mu \in \mathbb{R}^2\}$$

A similar representation of shape applies to a shape space of continuous parametrized curves $\beta : [0, 1] \rightarrow \mathbb{R}^2$.

$$[\beta] = \{(s\beta + \mu)R \mid R \in O(2), s > 0, \mu \in \mathbb{R}^2\}$$

Defining a proper metric for these (and other) shape spaces is a critical foundation for any type of statistical analysis of shape. By *proper* metric, I mean that it satisfies the mathematical definition of a metric and other sensible things as well. For example, the metric should specify that the distance between points within the same orbit is zero since they are regarded as the same shape. This will yield what I will call a *proper* shape distance on which a statistical shape analysis can be derived. By deriving a proper metric which is invariant to shape preserving transformations, the derived mathematical and statistical analysis will inherit those invariances so that the analysis is not plagued with artifacts of the data which are not relevant to the defined shape space. One such distance can be defined as follows,

$$d(C_1, C_2) = \min_{s \in [0, \infty), \mu \in \mathbb{R}^2, R \in O(2)} \|C_1 - (sC_2 + \mu)R\|.$$

In this case, if C_1 and C_2 are translated to the origin and scaled to length 1 as a preprocessing step, then this minimization can be solved by finding an optimal rotation (see “Functional and Shape Data Analysis” 2016 Anuj Srivastava and Eric P. Klassen). The optimal rotation can be solved using Procrustes analysis. To get the optimal rotation between C_1 and C_2 , take the singular value decomposition of $C_1^T C_2 \in \mathbb{R}^{2 \times 2}$ as

$$C_1 C_2^T = U \Sigma V^T$$

This decomposition can be accomplished using `numpy.linalg.svd`. The matrix $R = UV^T$ is an orthogonal matrix which will optimally rotate C_1 to align with C_2 . *I may have the order backwards i.e. it might be rotating C_2 to optimally align with C_1 .*

- translate and scale the first two curves in the shape database
- find the optimal rotation aligning one set of points to the other
- plot the aligned curves together

The Procrustes rotation can be used to approximate the rotational alignment for continuous curves as well. For piecewise linear I believe (but am not 100% certain) that the solution is the same.