IT575 Computational Shape Modeling **Assignment - 1**

Name - Vaibhav Amit Patel
ID No. 201401222

Name - Maharshi Vyas
ID No. 201401414

Date of Submission:

---

1. Show that the full Procrustes distance d_F is a metric on the shape space. Assume that $d([p],[q]) = \cos^{-1}(|\langle p_c, q_c\rangle|)$, where p_c ,q_c denotes the representation of any landmark configuration in the equivalence class $[p]$, $[q]$ respectively as a vector in $\mathbb{C}^n$, is a valid metric on the shape space.

   Answer : Let's denote the shape space to be V. Now, any metric on this shape space V is a function $d\colon V \times V \to \big[0,\infty\big]$. Where $\big[0,\infty\big]$ is the set of non-negative real numbers. d should satisfy following conditions:

   (a) $d\big(x,y\big) \geq 0$.
   (b) $d\big(x,y\big) = 0 \iff x = y$.
   (c) $d\big(x,y\big) = d\big(y,x\big)$.
   (d) $d\big(x,z\big) \leq d\big(x,y\big) + d\big(y,z\big)$.

   We will show that full Procrustes distance satisfy all of these conditions one by one.
   Assuming Kendall shape distance,

   $$\theta = d([p],[q]) = \cos^{-1}(|\langle p_c, q_c\rangle|)$$

   Full Procrustes Distance, $d([p],[q]) = \sin\theta$

   (a) $d\big(x,y\big) \geq 0$.

   $$d([p],[q]) = \sin(\cos^{-1}(|\langle p_c, q_c\rangle|))$$

   Range of $\cos^{-1}$ is $\big[0,\pi/2\big]$.Sine is nonegative function in first quadrant. So the metric d satisfies this condition.

   (b) $d\big(x,y\big) = 0 \iff x = y$.

   $$d_F([p],[q]) = 0$$
   $$sin(\cos^{-1}(|\langle p_c, q_c\rangle|)) = 0$$
   $$\cos^{-1}(|\langle p_c, q_c\rangle|) = 0$$

   (Sine has value 0 at 0)

   $$e^{i\theta} = 1$$
   $$\theta = 0$$

   $\implies$ $[p]$ and $[q]$ are in the same direction.

   Now, in shape space $[p]$ and $[q]$ are already made invariant to translation and scaling, the only thing left is rotation. And here, we are getting that between $[p]$ and $[q]$ angle is zero. So, there is only one conclusion to this which is that they are same.

(c) $d\big(x,y\big) = d\big(y,x\big)$.
It is the angle between the absolute value of the inner product.

$$|\langle p_c, q_c \rangle| = |\langle q_c, p_c \rangle|$$
$$\cos^{-1}(|\langle p_c, q_c \rangle|) = \cos^{-1}(|\langle q_c, p_c \rangle|)$$
$$\sin(\cos^{-1}(|\langle p_c, q_c \rangle|)) = \sin(\cos^{-1}(|\langle q_c, p_c \rangle|))$$

(Sine is a bijective function in first quadrant)

$$d_F([p],[q]) = d_F([q],[p])$$

(d) $d\big(x,z\big) \leq d\big(x,y\big) + d\big(y,z\big)$
Assuming, Kendall shape distance is a valid metric on the shape space.

$$\cos^{-1}(|\langle p_c, q_c \rangle|) \geq \cos^{-1}(|\langle p_c, r_c \rangle|) + \cos^{-1}(|\langle r_c, q_c \rangle|)$$

$$\theta_3 \geq \theta_1 + \theta_2$$

(Assuming a θ for each distance)

$$\sin\theta_3 \geq \sin(\theta_1 + \theta_2)$$

(Sine is strictly increasing in the first quadrant)

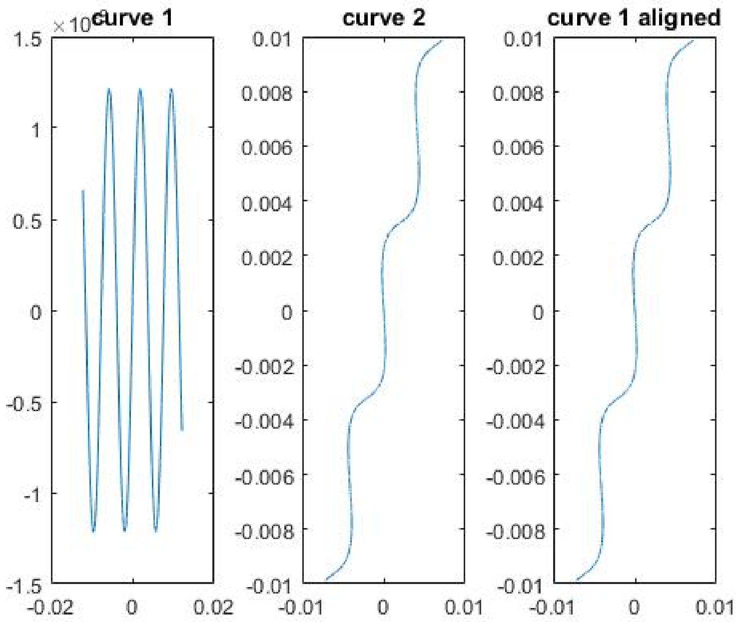
$$\sin\theta_3 \geq \sin\theta_1 \cos\theta_2 + \cos\theta_1 \; sin\theta_2$$

Now, $\cos\theta_2$,$\cos\theta_1 \leq 1$ because θ_1, θ_2 are in first quadrant. Hence,

$$\sin\theta_3 \geq \sin\theta_1 + \sin\theta_2$$

$$\sin(\cos^{-1}(|\langle p_c, q_c \rangle|)) \geq \sin(\cos^{-1}(|\langle p_c, r_c \rangle|)) + \sin(\cos^{-1}(|\langle r_c, q_c) \rangle|)$$
$$d_F([p],[q]) \geq d_F([p],[r]) + d_F([r],[q])$$

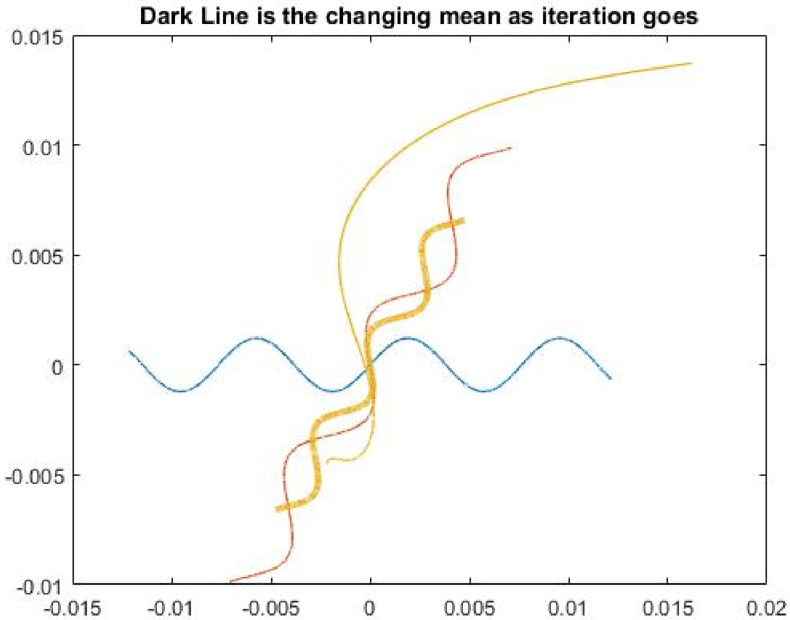
2. Write a *MATLAB* function *myProcrustesAlign.m*, to be used as $pa = myProcrustesAlign(p,q)$ that will align the set of landmarks p to landmarks in q. The inputs p,q and output pa should be $k \times 2$ matrices, where k is the number of landmarks.



We have used 18 landmark points for each image. Here we can see the actual variation between these landmarks points as the images vary.

Figure 1: **The variations in the landmarks**

3. Write a *MATLAB* function *myProcrustesAlign.m*, to be used as $mnshp = myProcrustesMean(p,q)$ that outputs the mean shape as a $k \times 2$ matrix, with the input p being a $k \times 2$ matrix for n objects represented as k landmarks. The x and y coordinates will be assumed to be put in alternating columns of matrix p. The mean shape should be computed using the iterative algorithm mentioned in class, wherein the iterations should stop once the norm of difference of successive mean estimates is below the threshold given as an input in the variable thr.



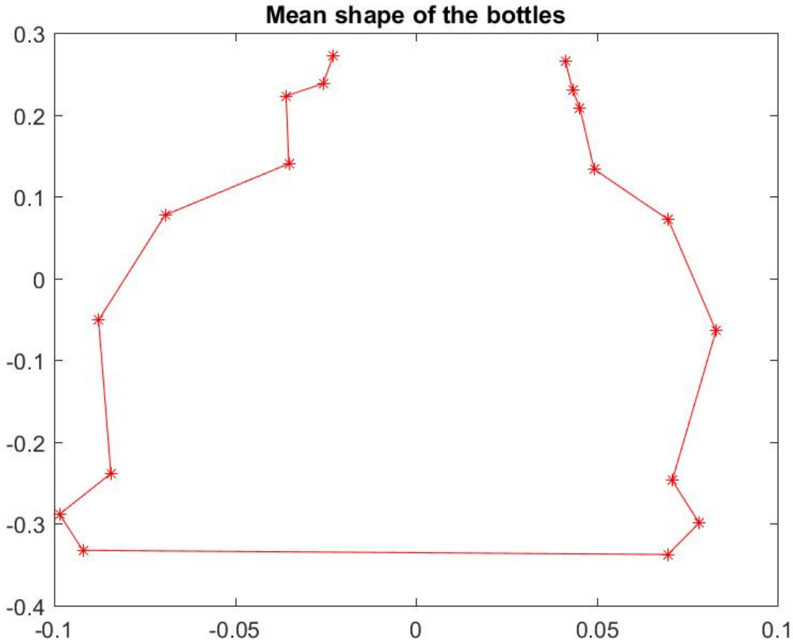
We have used 18 landmark points for each image. Here we can see the actual variation between these landmarks points as the images vary.

Figure 2: **The variations in the landmarks**

4. The shape of coke bottles has gone through a few changes over the years. You will find some representative images of these bottles in the folder bottles. For marking the *landmarkpoints*, use the MATLAB function **getpointsASM.m** . Type **help getpointsASM** to see its usage and related information. Use **imread.m** (available in MATLAB) for reading image files.

   (a) Justify the number of landmarks you will use to compute shape related informa-tion for these bottles. Compute and plot the mean shape of the coke bottle using myProcrustes-Mean.m.

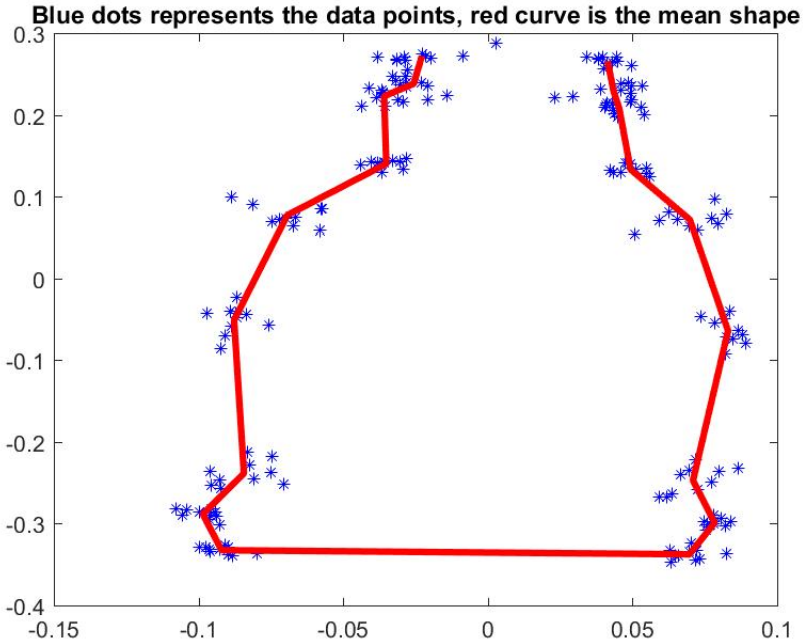
   Ans: We have taken a total of 18 landmark points to fit all the information in the bottles images. Landmarks should be oriented same with the edges in each image. We should consider an image with highest number of curves and edges to count number of landmarks and fit the same landmarks for all the images with corresponding segments.Thus, Shape information of each image is stored is optimize;



We have used 18 landmark points for each image. And we can see that these landmarks represents outline of a bottle very well.

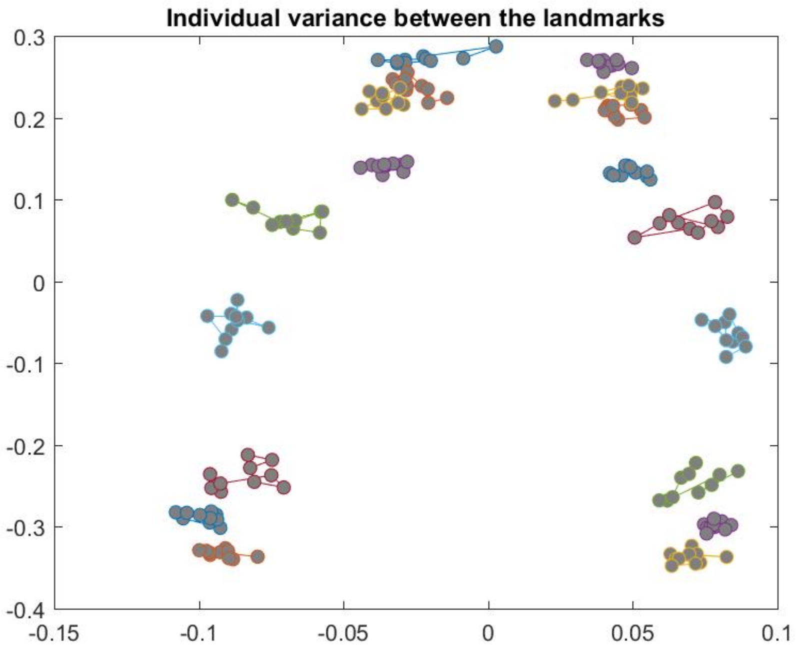
Figure 3: **Mean shape of 10 coke bottles.**

(b) Compute the shape covariance matrix and plot the eigenvalue profile. Find out the number of eigenvectors required to summarize 95% variance of the entire dataset. **Ans:** Eigenvalue profile is given in figure 7. According to the profile, 6 eigenvectors are required to summarize 95% variance of the wntire dataset..



Here blue dots are the total 180 points ($18 landmarks \times 10 bottles$) . We can see the change between these landmark points.

Figure 4: **All the landmark points and the mean fitting these points.**



We have used 18 landmark points for each image. Here we can see the actual variation between these landmarks points as the images vary.

Figure 5: **The variations in the landmarks**

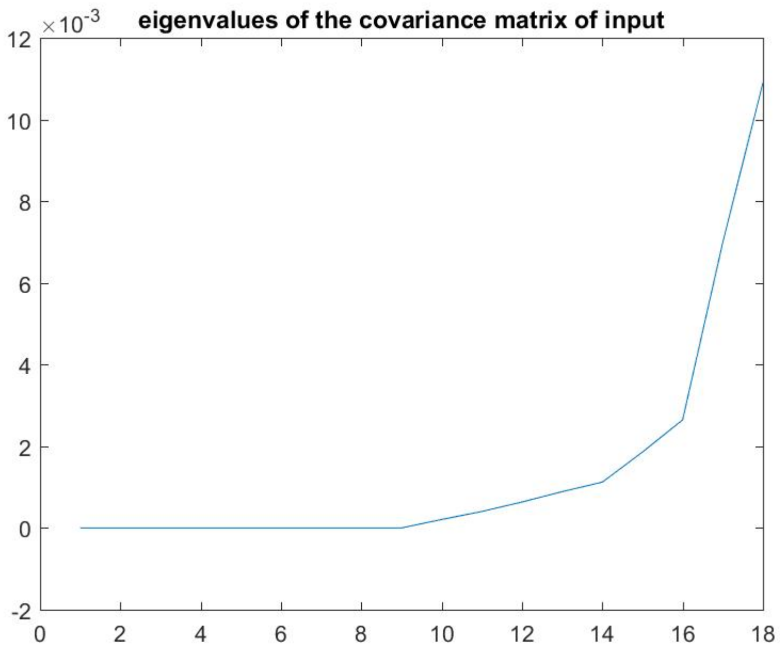


Figure 6: **The eigenvalue profile**

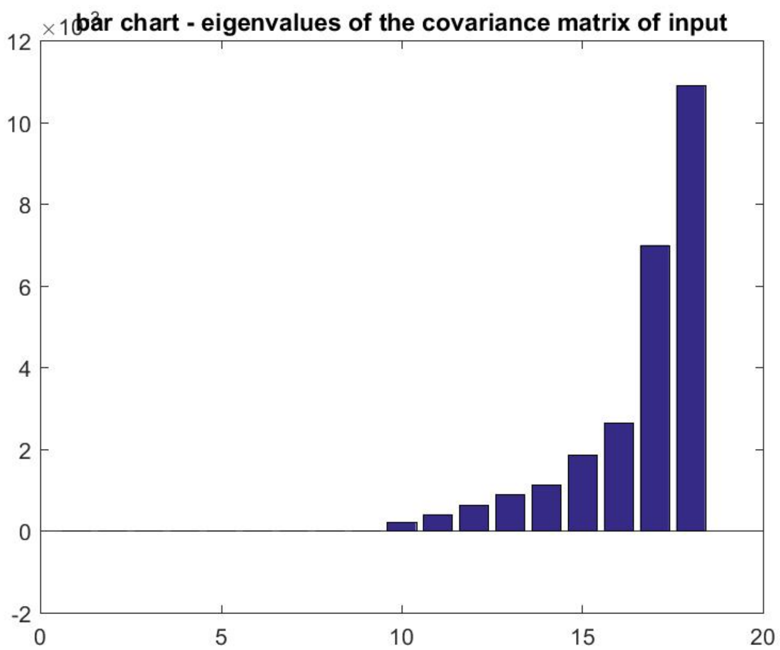
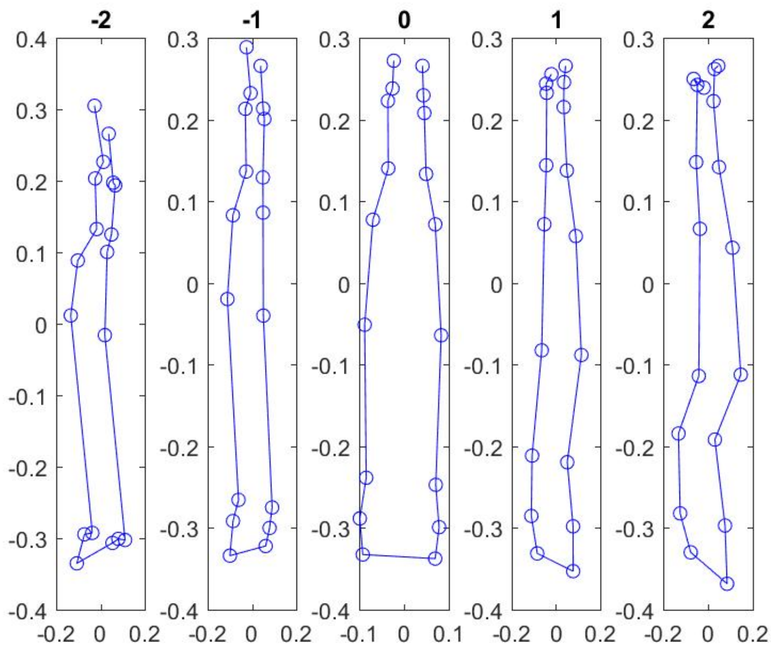


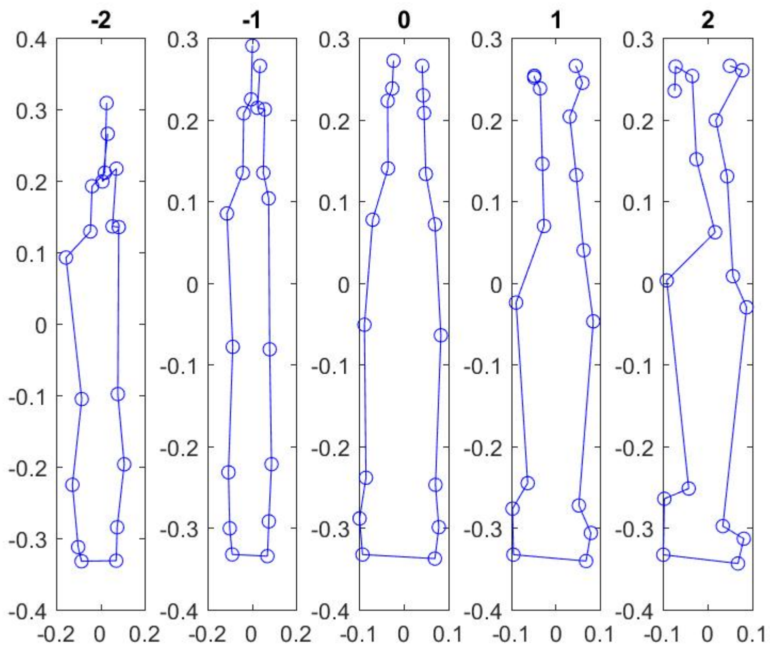
Figure 7: **The eigenvalue profile - bar chart**

(c) Let v 1 and v 2 be the first two principal components(eigenvectors). Plot variations in the mean shape along v 1 and v 2 , i.e, $+kvi$, where µ is the mean shape, $k[2,1,0,1,2], i = 1,2,$ and is an appropriate scalar.
Ans: Figure 5 shows landmark point cloud which shows how far landmarks can vary from its own set of landmark points. Figure 8 and 9 contains images with variations in the mean shape along largest and second largest eigenvector(The direction of highest variance or eigenvalue)



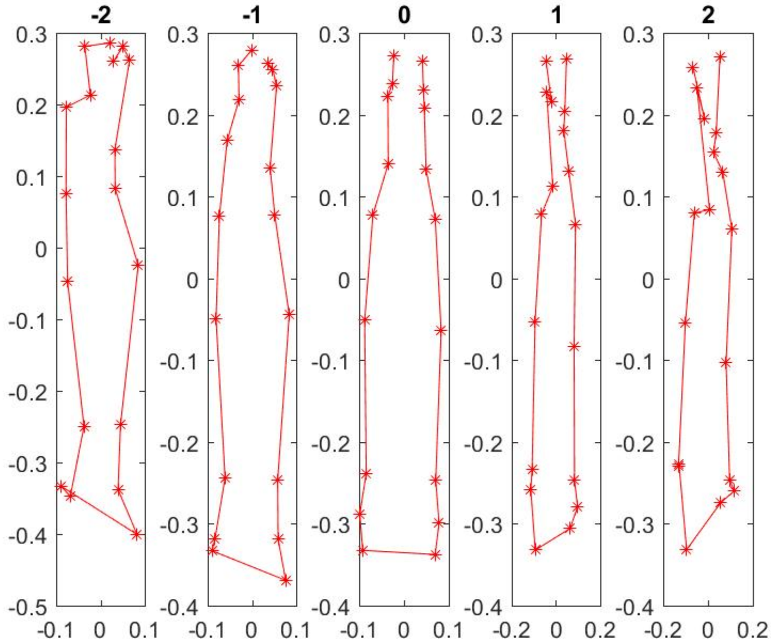
The title is the transformation constant k. We have transformed the mean shape using this formula: $\mu + k\Delta v_1$.

Figure 8: **Variation along Largest "eigenvector" (eigenvalue)**



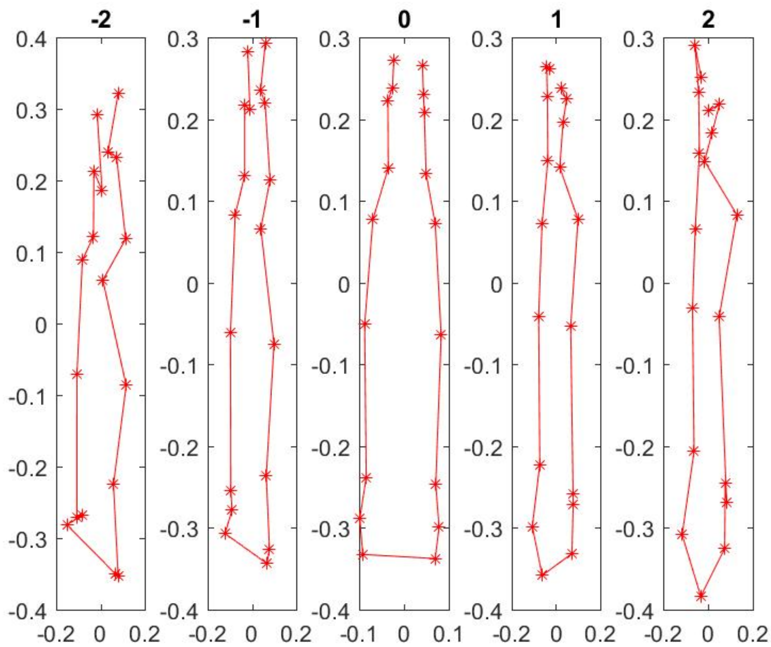
The title is the transformation constant k. We have transformed the mean shape using this formula: $\mu + k\Delta v_2$.

Figure 9: **Variation along second Largest "eigenvector" (eigenvalue)**



Here, we first converted the mean shape in the different basis (Same as in PCA) and then applied the transformation. After this we reverted back it to the original basis. We have transformed the mean shape using this formula: $\mu + k\Delta v_1$.

Figure 10: **Variation along Largest "eigenvector" (eigenvalue) in the PCA basis**



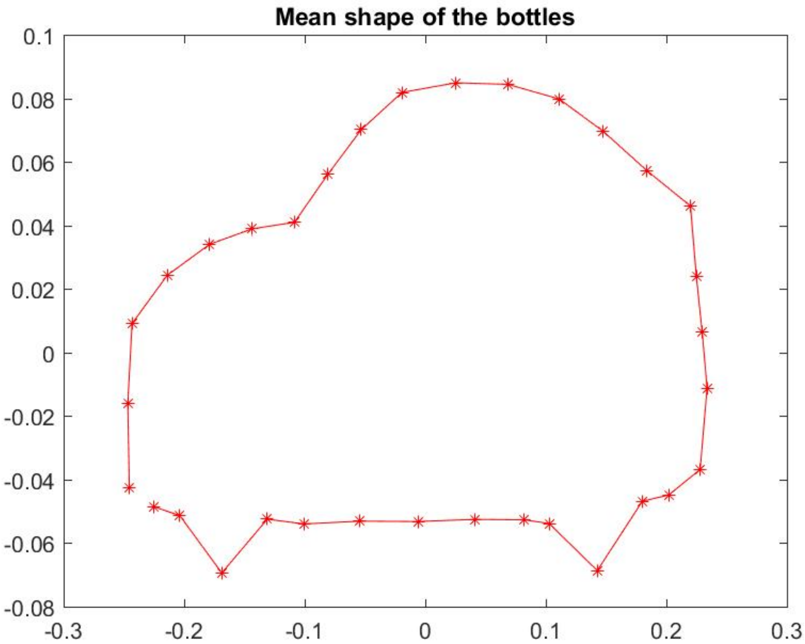
Here, we first converted the mean shape in the different basis (Same as in PCA) and then applied the transformation. After this we reverted back it to the original basis. We have transformed the mean shape using this formula: $\mu + k\Delta v_2$.

Figure 11: **Variation along second Largest "eigenvector" (eigenvalue) in the PCA basis**

5. Repeat the above exercise for car shape using images provided in the folder cars. If you think

that only the input changes from bottle landmarks to car landmarks, you are missing something.

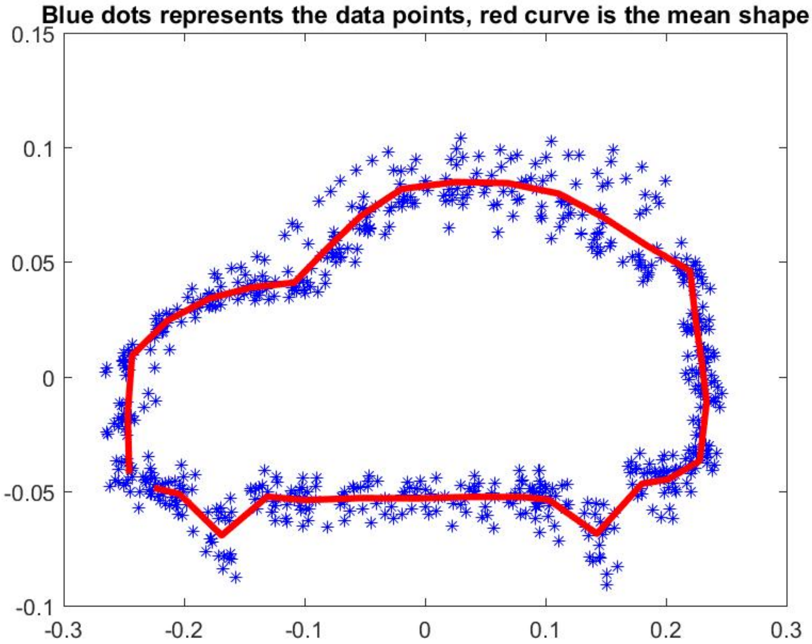
(a) Justify the number of landmarks you will use to compute shape related informa-tion for these cars. Compute and plot the mean shape of the car using myProcrustesMean.m.



We have used 33 landmark points for each image. And we can see that these landmarks represents outline of a car very well.

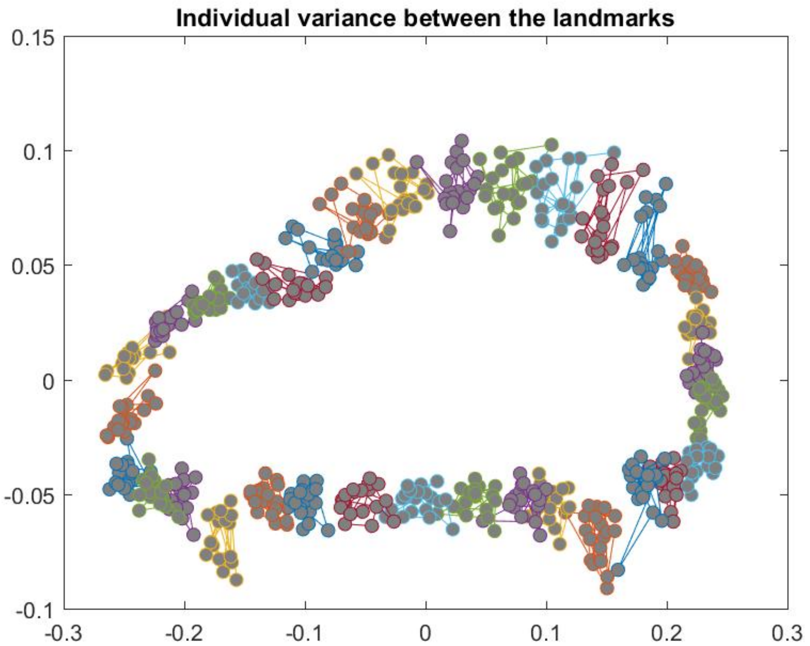
Figure 12: **Mean shape of 20 cars.**

(b) Compute the shape covariance matrix and plot the eigenvalue profile. Find out the number of eigenvectors required to summarize 95% variance of the entire dataset.



Here blue dots are the total 660 points ($33 landmarks \times 20 cars$) . We can see the change between these landmark points.

Figure 13: **All the landmark points and the mean fitting these points.**



We have used 33 landmark points for each image. Here we can see the actual variation between these landmarks points as the images vary.

Figure 14: **The variations in the landmarks**

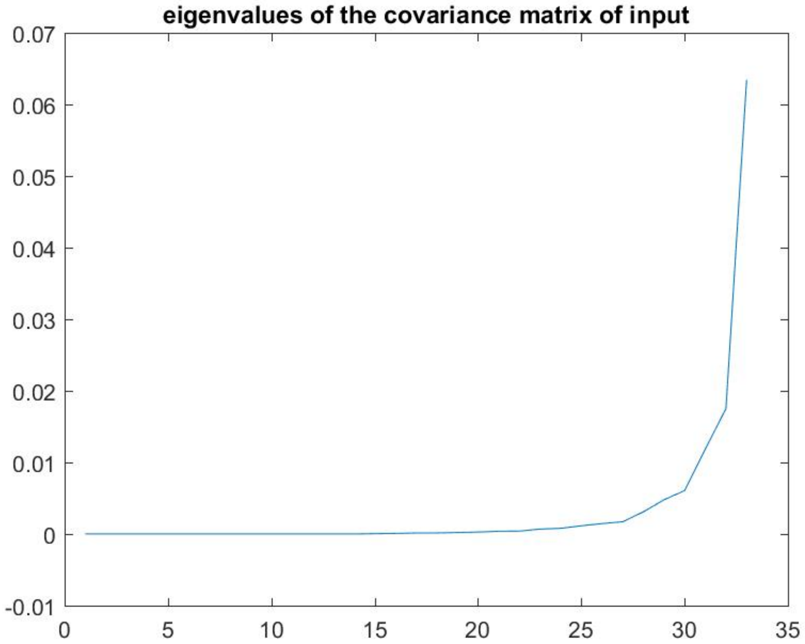


Figure 15: **The eigenvalue profile**

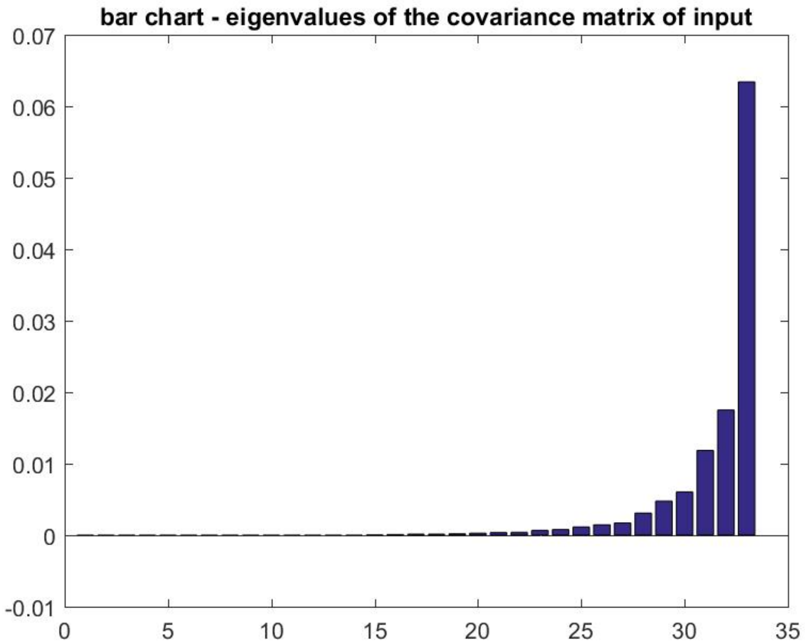
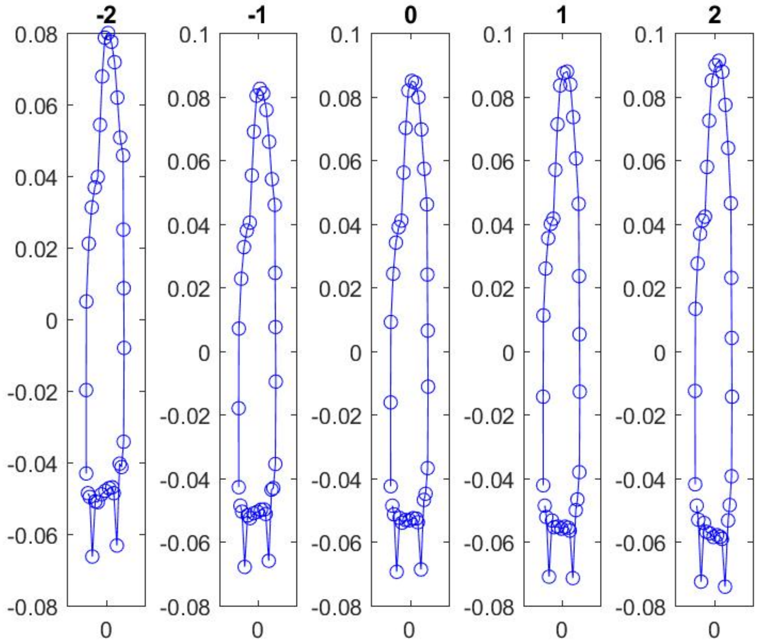


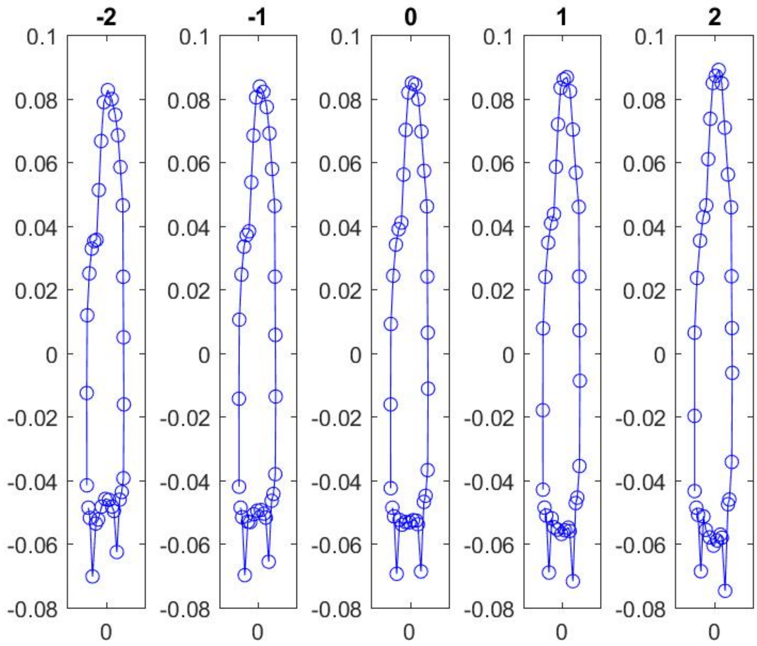
Figure 16: **The eigenvalue profile - bar chart**

(c) Let v 1 and v 2 be the first two principal components (eigenvectors). Plot variations in the mean shape along v 1 and v 2 , i.e, $+kvi$, where µ is the mean shape, $k[2,1,0,1,2], i=1,2$, and is an appropriate scalar.



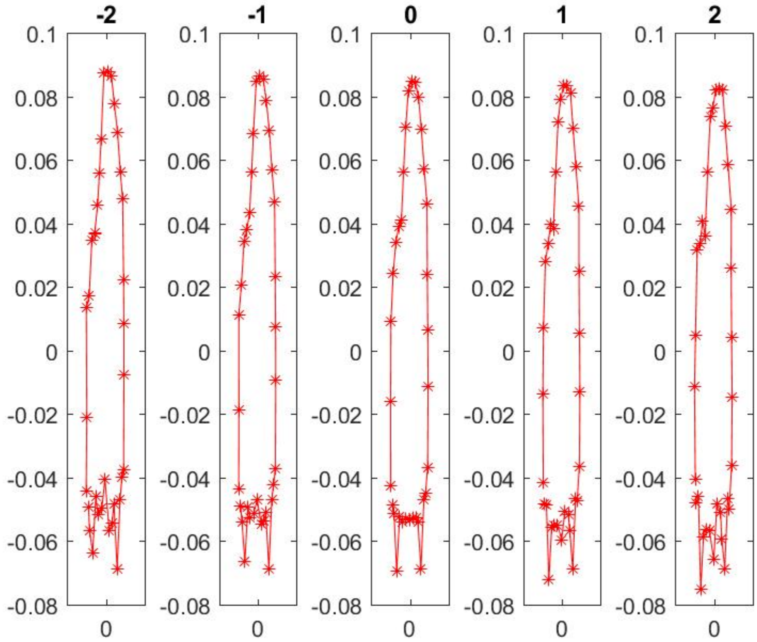
The title is the transformation constant k. We have transformed the mean shape using this formula: $\mu + k\Delta v_1$.

Figure 17: **Variation along Largest "eigenvector" (eigenvalue)**



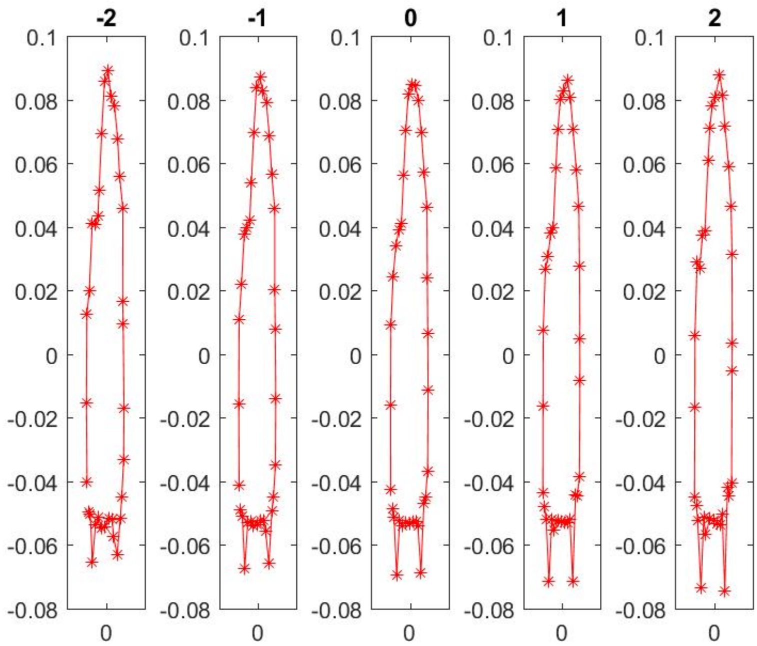
The title is the transformation constant k. We have transformed the mean shape using this formula: $\mu + k\Delta v_2$.

Figure 18: **Variation along second Largest "eigenvector" (eigenvalue)**



Here, we first converted the mean shape in the different basis (Same as in PCA) and then applied the transformation. After this we reverted back it to the original basis. We have transformed the mean shape using this formula: $\mu + k\Delta v_1$.

Figure 19: **Variation along Largest "eigenvector" (eigenvalue) in the PCA basis**



Here, we first converted the mean shape in the different basis (Same as in PCA) and then applied the transformation. After this we reverted back it to the original basis. We have transformed the mean shape using this formula: $\mu + k\Delta v_2$.

Figure 20: **Variation along second Largest "eigenvector" (eigenvalue) in the PCA basis**