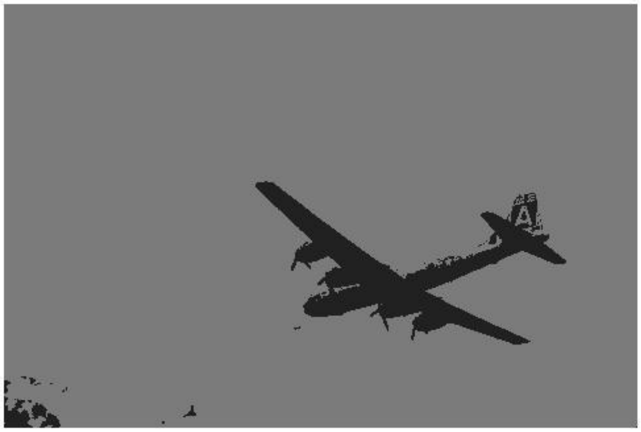
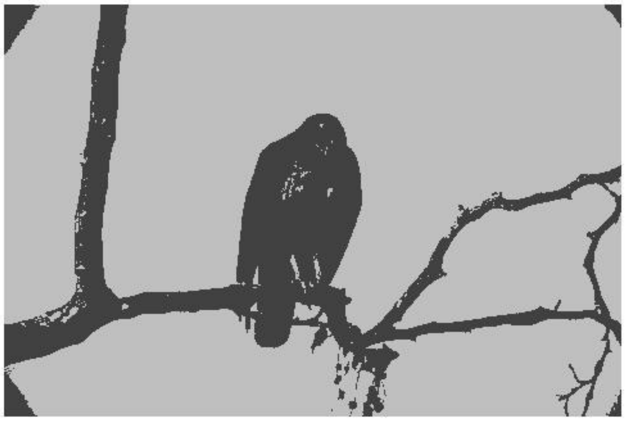
1. Image segmentation with K-means clustering and GMM clustering
   With K-mean clustering, test with K = {2,3,4,5} to generate the image segmentation based on RGB colors and coordinate.
   Contrast color of enhancement to grayscale to make the visualization be clear to observe.

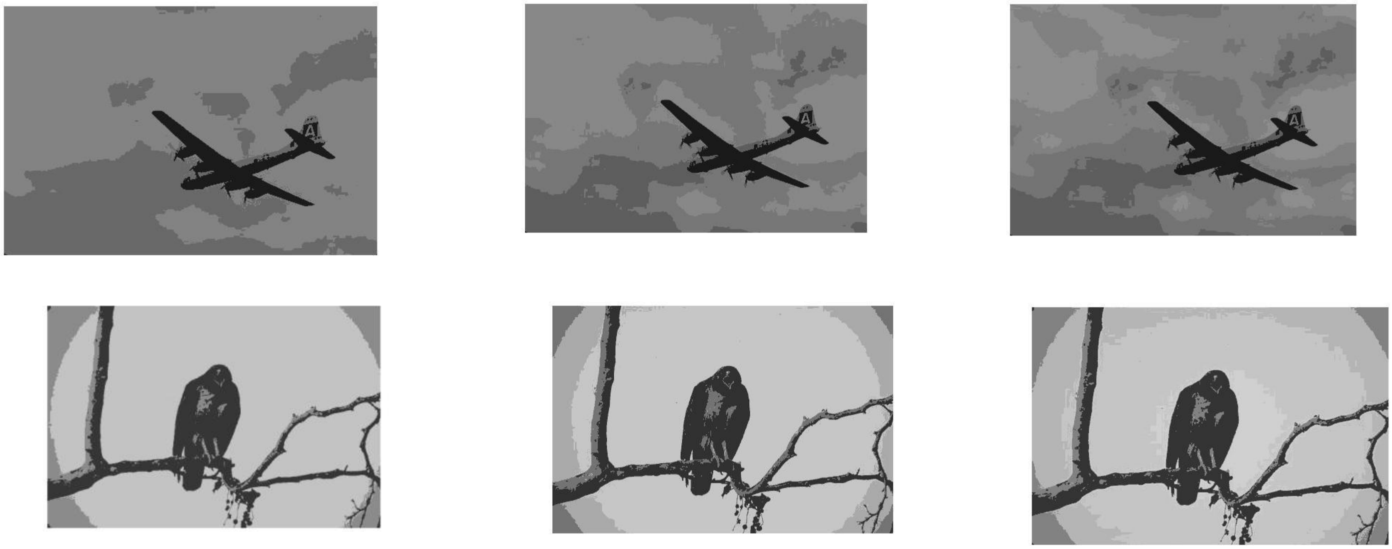
The original pictures has the same features that both have a plant area show the similar color. But the objects (the plane and the bird) has a different color area. We could guess the object could be showed when after doing the clustering (image segmentation). The following will show the result of K-means clustering and the GMM clustering result.

K-means clustering has done with 5 dimensional data, where 1st and the 2nd data are the coordinate data and the 3rd 4th 5th are the red, green, blue data. Since the clustering algorithm take the RGB data as the most calculating data. The coordinate data could be replace by pixels ordering. On 321*481 pixels image, take the pixels order from 1 to 321*481 = 154401. Then doing the K-means algorithm to calculate the distance and the labels of each pixel.



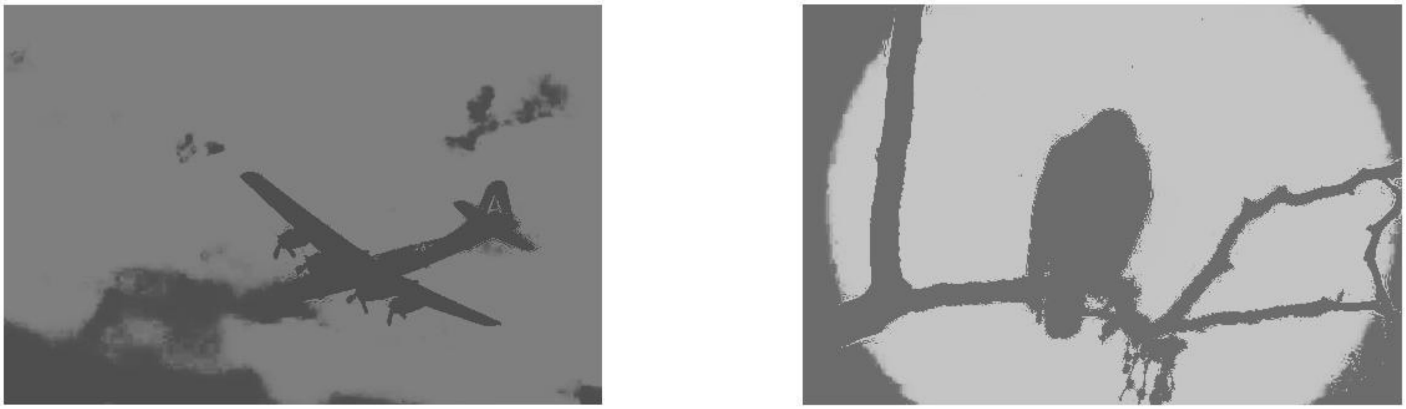


Two picture shows the K-means clustering method (K = 2) done on plane and bird's image. We could observe that the images are showed as two different gray sale after the K-means algorithm. The plane and the bird's shape is visualized in two images.

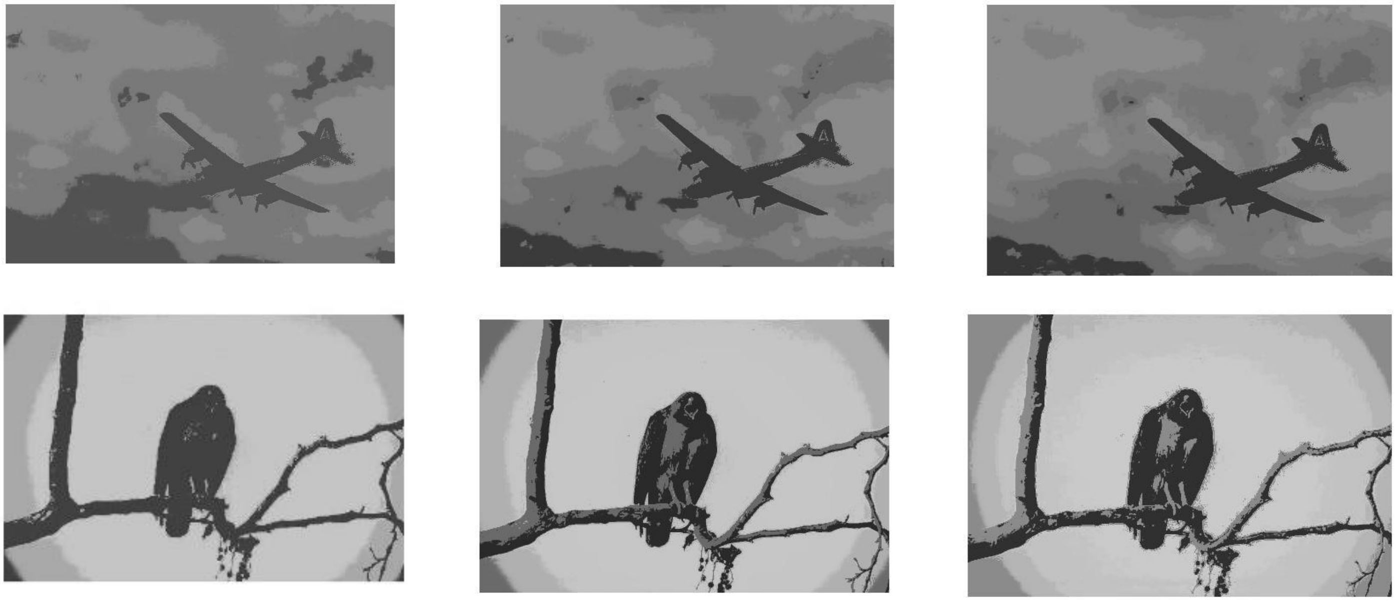


Picture above shows the result of K-means clustering result. Left K = 3, Middle K = 4, Right K = 5. As the value of K increasing, the image shows more gradually on grayscale.

GMM clustering segmentation. The following results will show the GMM based image segmentation. With applying the EM algorithm on each pixel, we could find the on each pixels of K components.

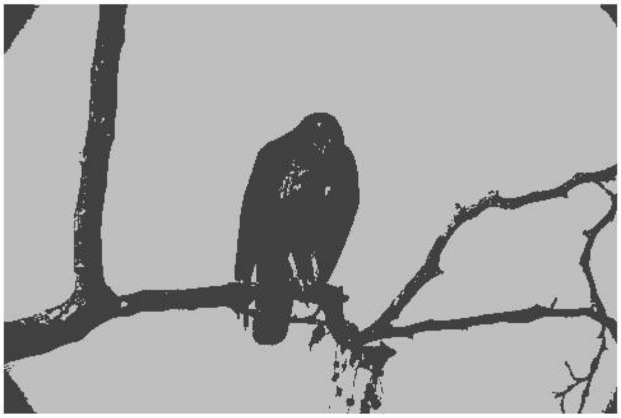


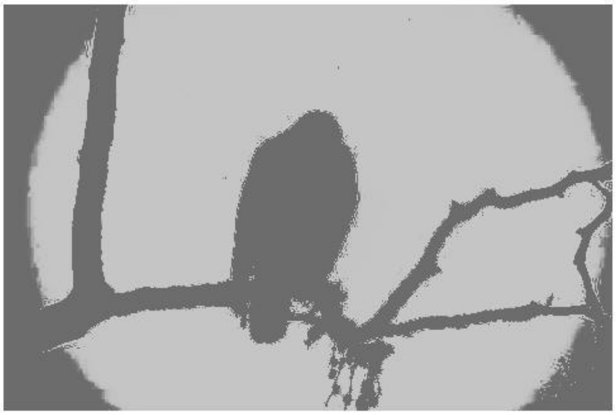
Two pictures above shows the result of GMM clustering segmentation (with component number = 2). The object’s shape is visualized as well by two different grayscale. But some bad result appear in this two pictures. For instance, the head of the plane is not showing well in GMM clustering. Some of the cloud are decided as the same scale as plane.



The pictures above show that the result of GMM based clustering. As the value of K increasing (number of components). The showing results are getting better.

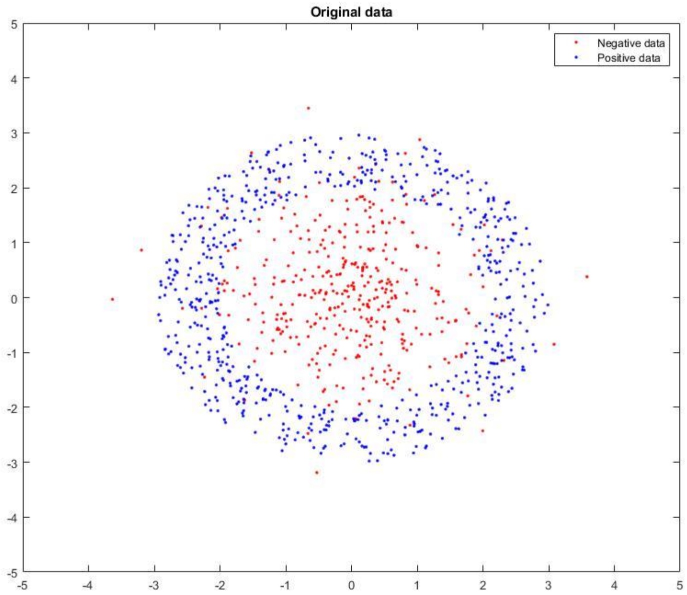
Difference of K-means clustering and GMM based clustering:





Here we compare the result of K-means clustering and the GMM based clustering result.
The left picture is the K-means clustering as K = 2 and the right picture is the GMM based clustering as K-components K = 2. Here we can observe that the shape of the bird is detected as well enough. However, on the detail of the bird K-means clustering shows more. The GMM based one did not show the light color of the bird. These results show that the K-mean took each pixel distance to decide the label. But the GMM based clustering on each pixel might influenced by the neighbor pixels.

2. Train two SVM classifier, the linear SVM classifier and the Gaussian SVM classifier. The data from class −1 are drawn from a Gaussian with zero-mean and identity covariance-matrix. The data from class +1 are generated using a two-step procedure: a radius value is drawn from a uniform distribution over the interval [2,3] and an angle value (in radians) is drawn from a uniform distribution over the interval [−π,π]. And the +1 data are converting to Cartesian coordinates. Showing the two-dimension data on the plot.



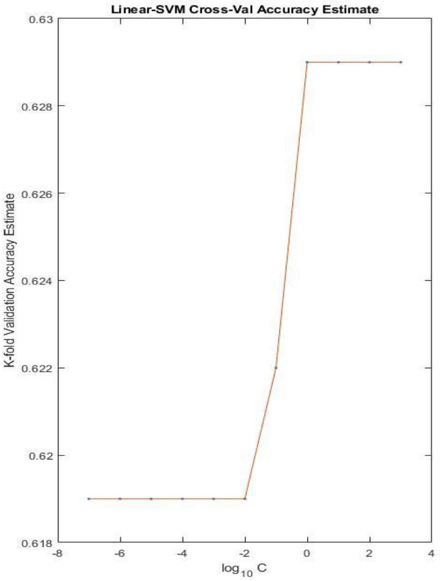
This plot shows the original data of 2 data sets.

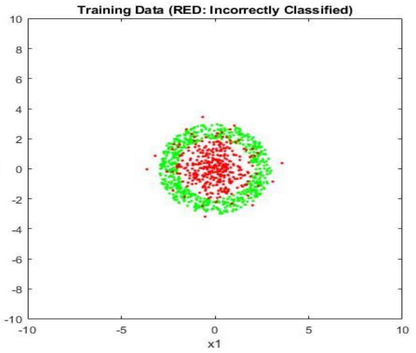
Red points are the two Gaussian mixture data with mean values = [0,0], sigma = [1 0;0 1];

Blue points are the data generated by define the uniform distribution random over interval [2,3] a s radius, and uniform distribution random over [-pi,pi] as angle.

Now we use these data to train SVC linear and SVC Gaussian classifier.

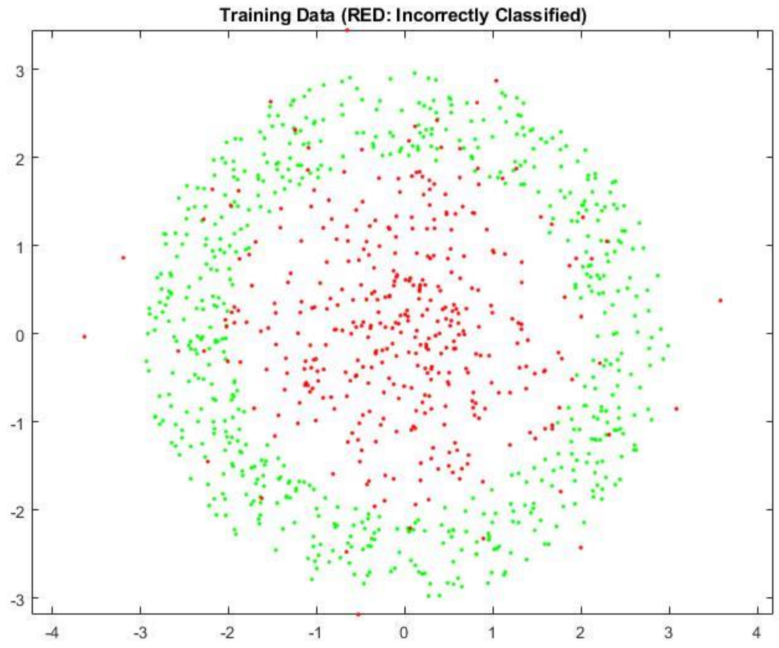
Linear SVM classifier:
Adjust the hyperparameter (make CList for different C value to train the classifier)
Set C of linear SVM classifier have 11 value for getting the best C value. C value from 10^-1 to 10^9
The training result is as below





We got the result that the err or equal to the negative data. Since the linear classifier could not work on the non linear data. Any decision could lead to high error rate except the circle boundary.

Error rate = 0.38 = 38%



Same situation appears on the testing data. Linear SVM classifier could not use on the data which generate by distribution on radius and angle 2 dimension data.

Error rate 0.332 = 33.2%

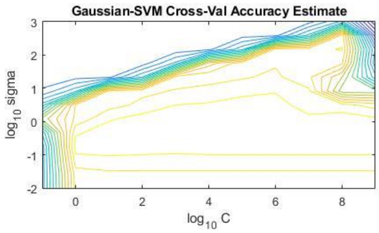
Gaussian SVM classifier:

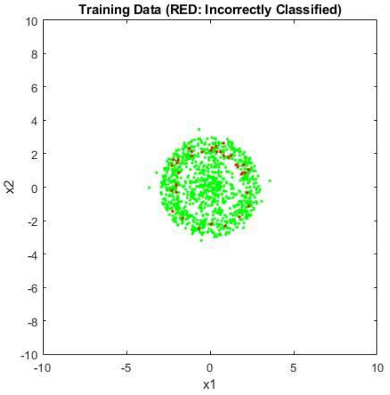
Now doing the same thing for the Gaussian classifier. Adjust the C List value for training the classifier.

Set C of linear SVM classifier have 11 value for getting the best C value. C value from 10^-1 to 10^9

The training result is as below

The Gaussian SVM classifier trained with the data and the result shows a better classification than linear one.

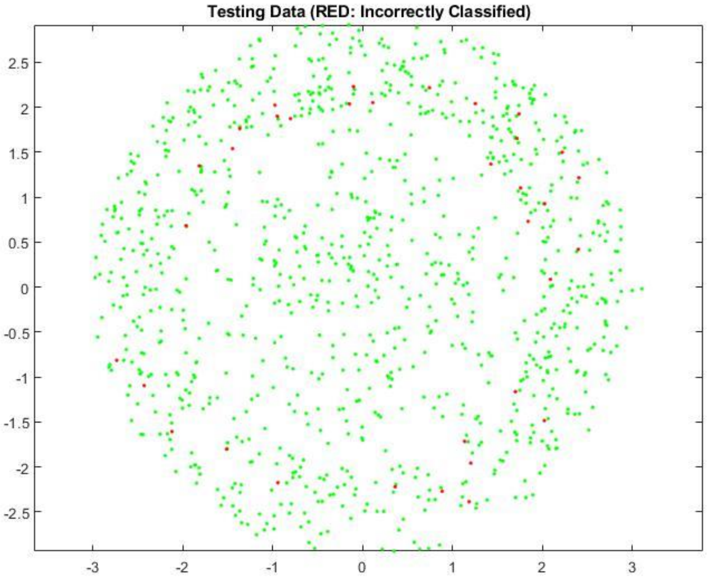




Since Gaussian K mean is on of non-linear classifier, the error of the training is equal to 0.037 = 3.7%.

Here we the best C = 10^7 for testing the individual testing data.

The contour shows each fold training for finding the best result od C value where has the largest correct label on positive data.



Here we found the result of applying best C value on individual testing data. The error rate = 0.034 = 3.4% which is similar as the training result.

In conclusion, the linear SVM classifier will only works on the case that all of the data are generate as 2-dimension data. Not working on the case that data generated via Polar coordinate. If the polar coordinate is shown on 2 axis separately for data (which mean the value is not converted to Cartesian coordinates). It might be work.

Fortunately, the Gaussian SVM classifier works well for each case, because the non-linear boundary will choose a better decision.

Code resource:

https://github.com/MakiseYuki/EECE5644-Machine-Learning/tree/master/Homework%204