Nguyen, Theodore

704-156-701

Math 156 Spring 2016

Lab 6: K-means

1. (5) Complete the function `km.m` which performs the k-means iteration on an arbitrary collection of points in $\mathbb{R}^n$. If the data is denoted by x_i and we are trying to find clusters c_j, in each iteration your code should assign each observation to the nearest centroid:

$$k_i \leftarrow \arg\min_j ||c_j - x_i||_2 \quad \forall i$$

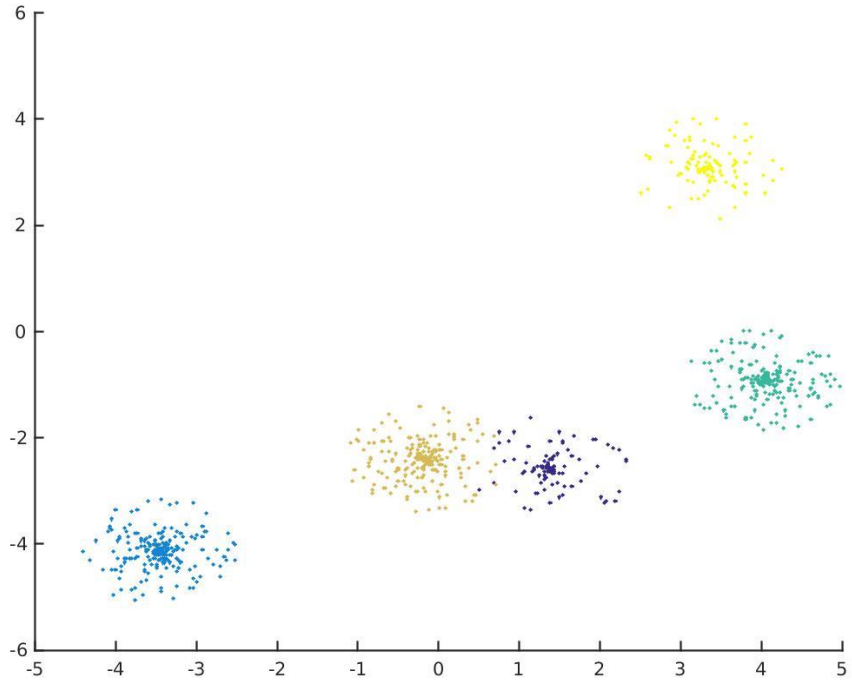
and then re-learn the centroids based on an average of assigned data points:

$$c_j \leftarrow \frac{\sum_{i:k_i=j} x_i}{\sum_{i:k_i=j} 1} \quad \forall j$$

where the index i is always assumed to range over the number of observations and j over the number of clusters being learned. The initial parameters for each cluster should be chosen by randomly selecting data points as centroids, and the algorithm should run until convergence (that is, data points no longer change assignment).

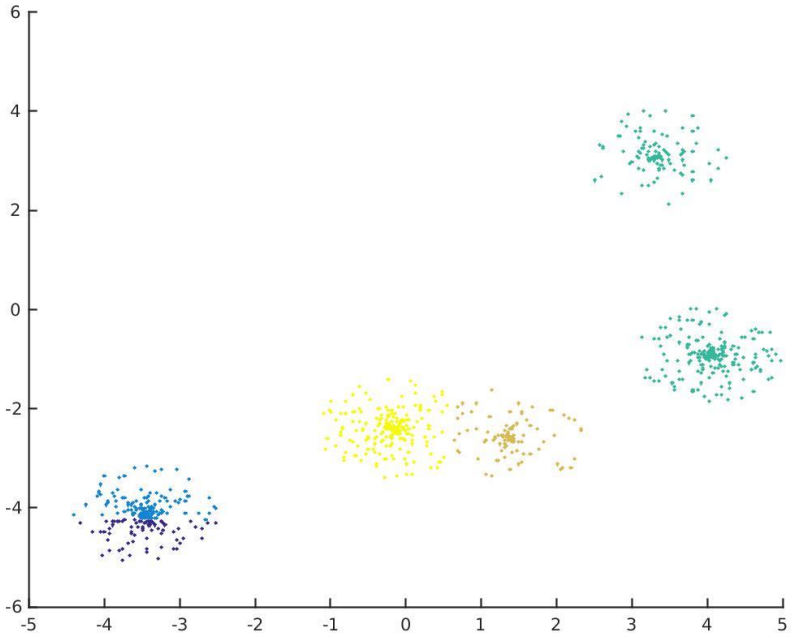
- Part 1 is attached as km.m

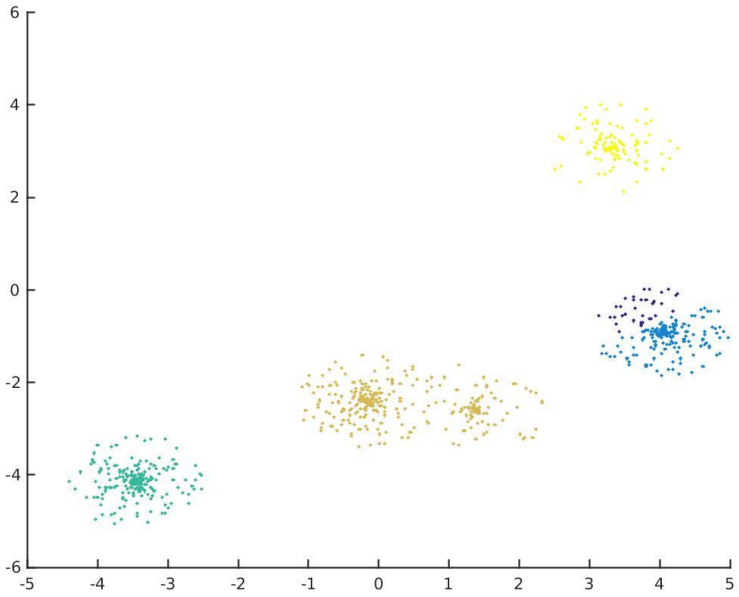
2. Examine the data generated by pointclouds.m. In particular, make sure to look at a scatter plot of the data for example by using **scatter(X(1,:),X(2,:),1,Y).** Finally, use the k-means function to classify the data generated by pointclouds.m into five clusters.

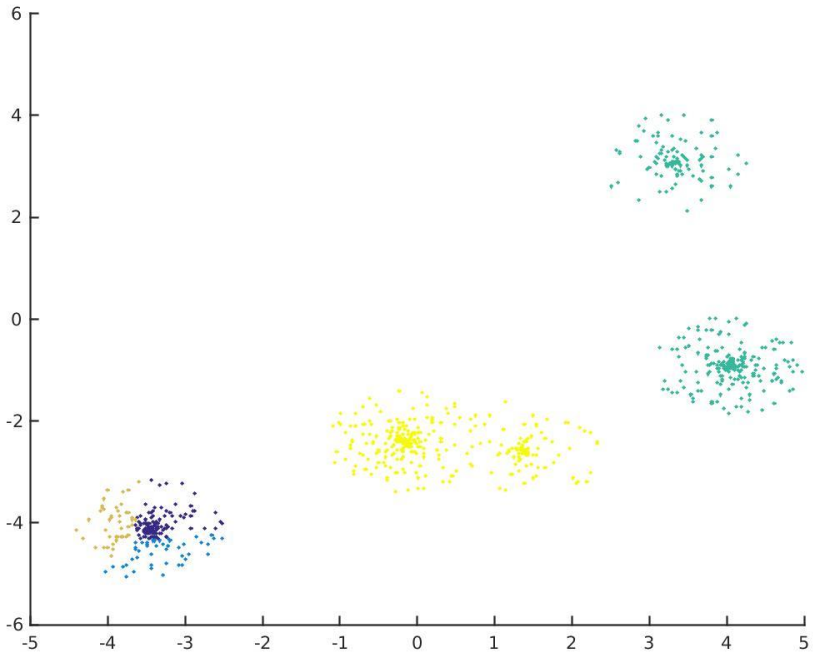


- The latter plot is attached as pointclouds.jpeg. It shows the plot of the generated point clouds, and their true cluster coloring

3. (4) Generate and turn in the classification result of your code for three separate runs of the k-means algorithm. This should be in the form of three scatter plots with the class assignments visually distinguished.



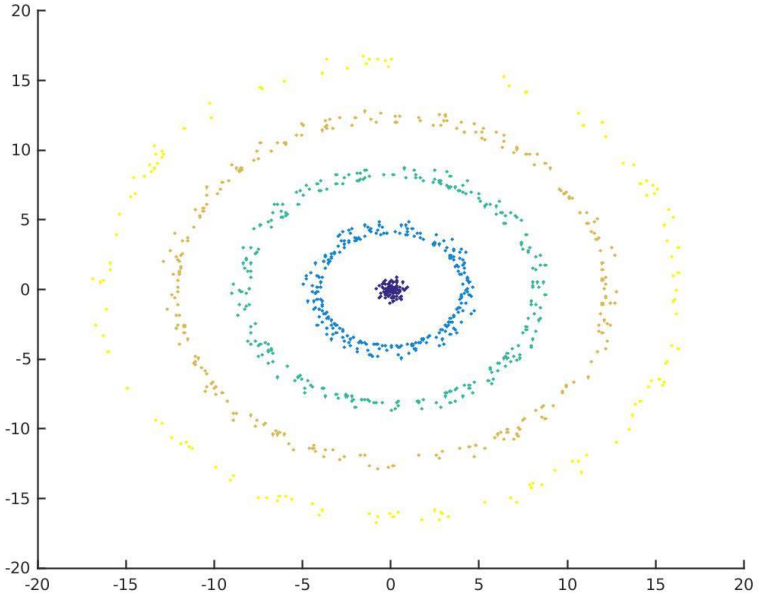




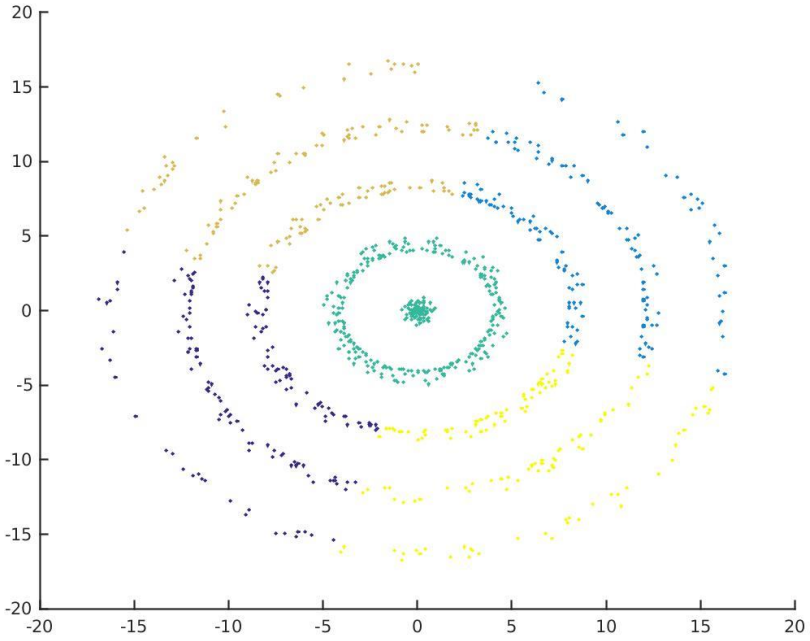
- The latter plots are attached as fc1.jpeg, fc2.jpeg, and fc3.jpeg; they are the plot of generated point clouds, with the colored classification classified by the K-means algorithm implemented.

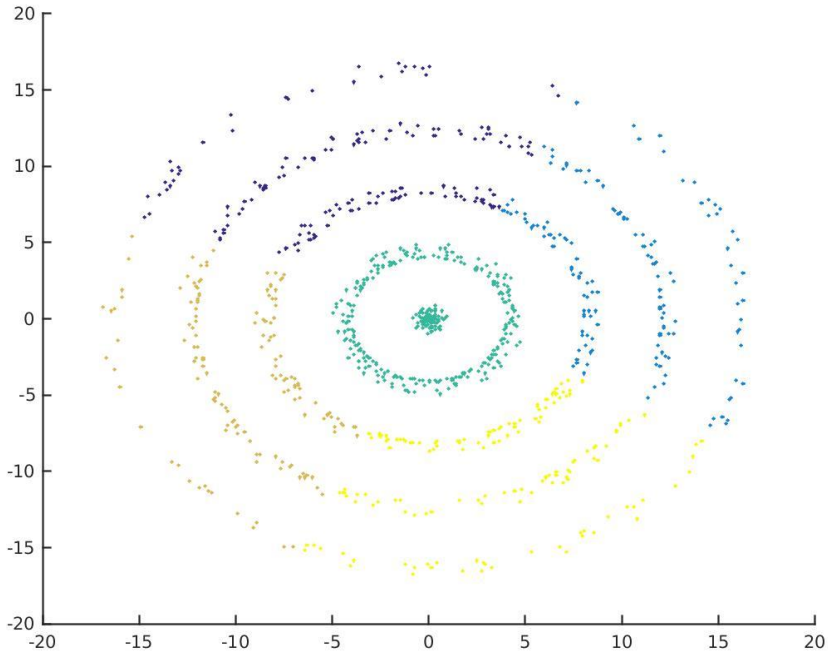
4. (4) Do the same for the pointrings.m dataset, again producing three distinct classification result figures.

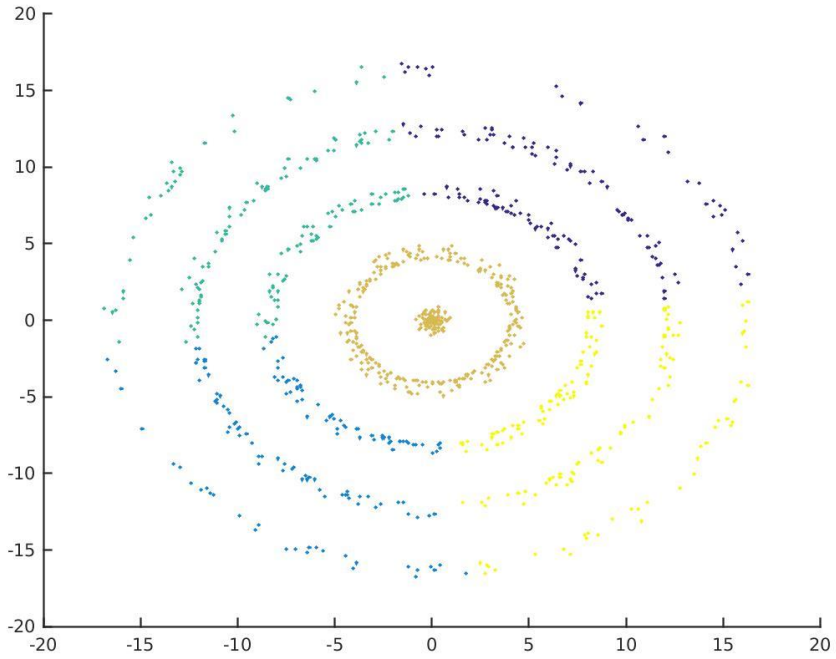
- The following is the true classification of the generated ring points;



- The following are attached as fr1.jpeg, fr2.jpeg, and fr3.jpeg and are the plots of the ring-generated data points with the K-means classification colors







5. (2) Which dataset do you believe k-means performed a better job clustering, on average? Why do you believe this is the case?

- I believe K-means does a better job clustering the cloud-based dataset instead of the ring-based. This is because K-means clusters based off the distance of points; in the cloud-based set, all the data points that are truly in the same clusters are all grouped close together with regards to distance. In the ring-based, the points are grouped based off of their distance from the center of the ring, rather than their distances from each other. The fact that K-means classifies solely off the distance would render it extremely unlikely to classify such points.

6. Use the function **im2rgb.m** to read the pixel RGB values of either (pick one) provided image. Perform a k-means clustering of these color values in R_3 using ten clusters.

7. (5) Generate and turn in the image produced by assigning each pixel to the centroid of the class it belongs to. This can be accomplished with the command:
imshow(rgb2im(C(:,labels), dims))
where **C** contains the learned centroid values, labels contains the learned class assignments, and **dims** is the image dimensions from **im2rgb.**

- I operated on the provided “mountains_small.png”. The original image and the image that was the output of the process is displayed below. The generated image is attached as “image.jpeg”.

ORIGINAL IMAGE:

GENERATED IMAGE: