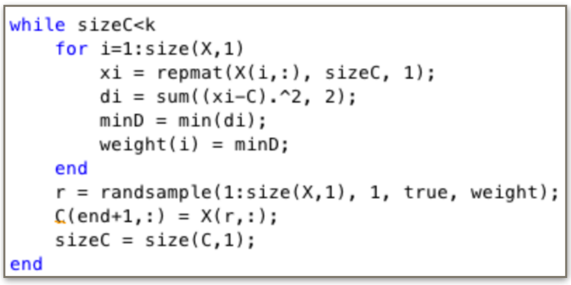
資工系100062236林修安

Machine Learning Assignment5 Report

Implementation

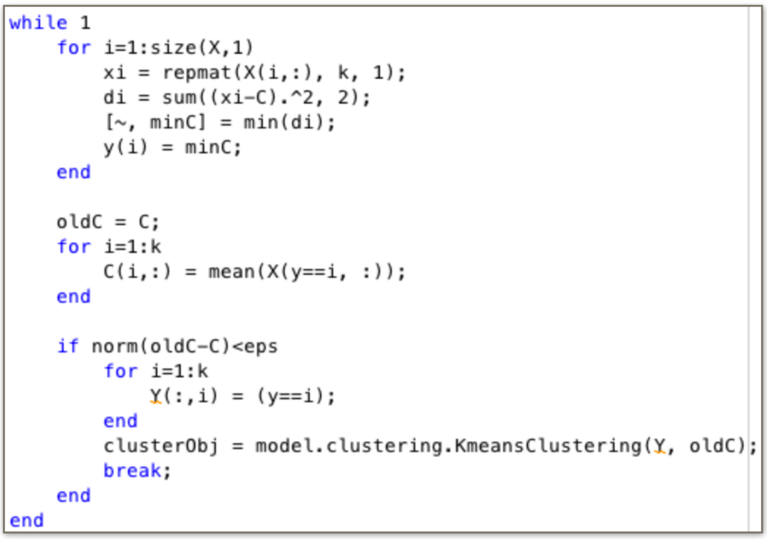
1. KmeansClustering

- Kmeans++ Initialization
  - Choose a random instance from the data as the first center
  - Compute $d(x)$, the distance between instance x and the nearest center
  - Choose the next center according to the weighted probability $\frac{d(x^{(t)})^2}{\sum_{t=1}^{N} d(x^{(t)})^2}$
  - Terminates until k centers are assigned



```
while sizeC<k
    for i=1:size(X,1)
        xi = repmat(X(i,:), sizeC, 1);
        di = sum((xi-C).^2, 2);
        minD = min(di);
        weight(i) = minD;
    end
    r = randsample(1:size(X,1), 1, true, weight);
    C(end+1,:) = X(r,:);
    sizeC = size(C,1);
end
```

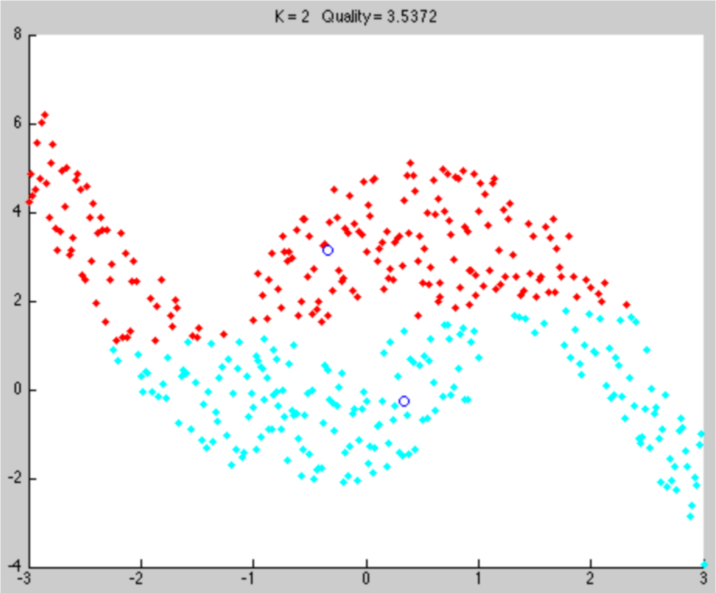
- `clustering(X,k)`
  - For each instance, calculate the distance to every center, and classify the instance to the cluster of the center with minimal distance
  - Calculate a new center for each class by taking mean of all instances in such class
  - Terminate if the difference of current centers and new centers is less than `eps`
  - Modify the output to the format as required

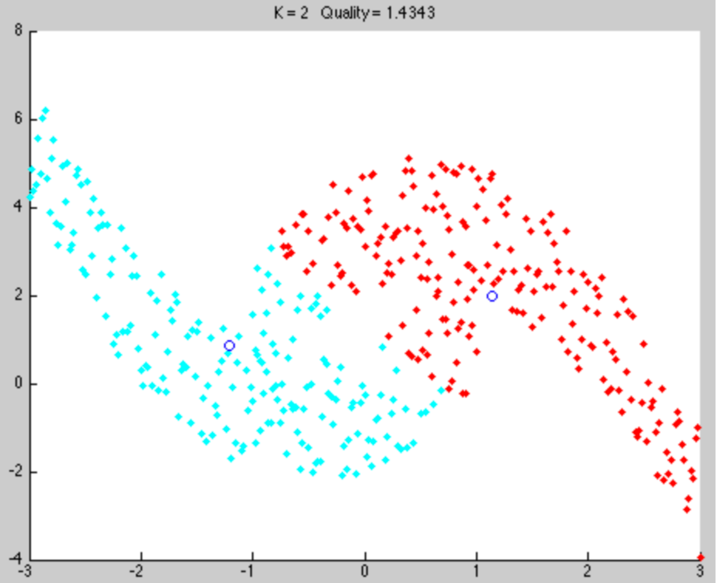


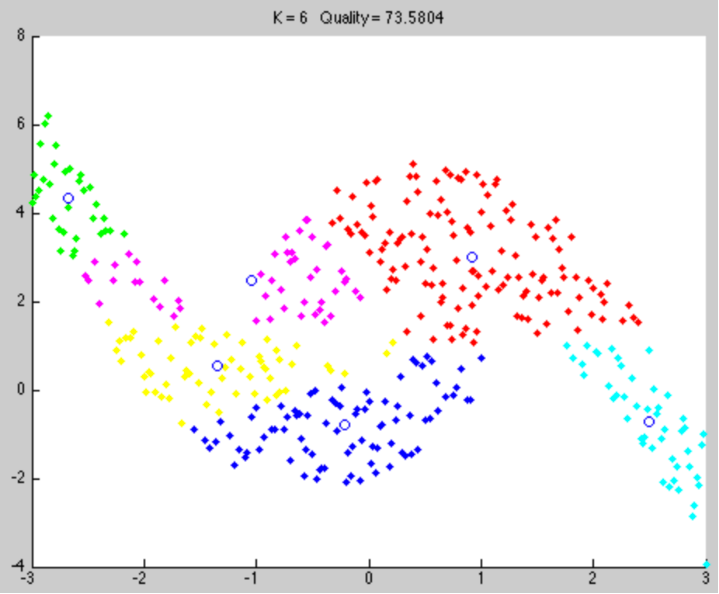
```
while 1
    for i=1:size(X,1)
        xi = repmat(X(i,:), k, 1);
        di = sum((xi-C).^2, 2);
        [~, minC] = min(di);
        y(i) = minC;
    end

    oldC = C;
    for i=1:k
        C(i,:) = mean(X(y==i, :));
    end

    if norm(oldC-C)<eps
        for i=1:k
            Y(:,i) = (y==i);
        end
        clusterObj = model.clustering.KmeansClustering(Y, oldC);
        break;
    end
end
```

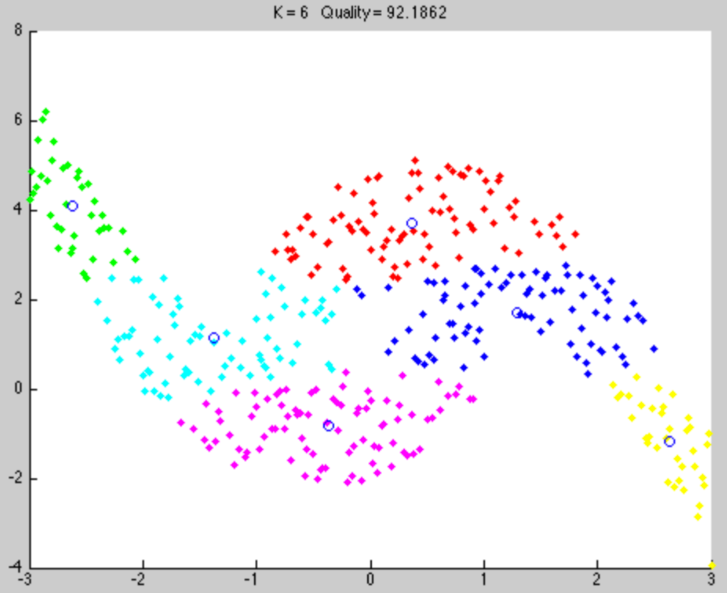
- Result

  - The blue circles are the centers of clusters
  - For k=2, the right result below seems more reasonable than the left one, and such difference is caused by different choices of initial prototypes
  - For k=6 in the left result below, the purple cluster seems odd because some of them should be green and some be yellow according to human eyes



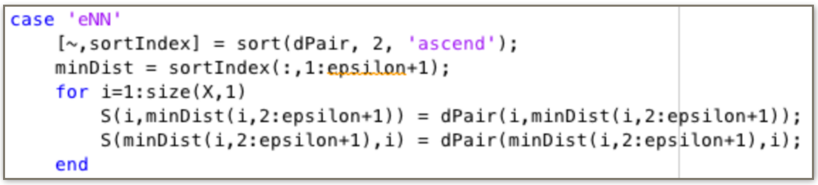






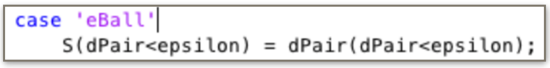
2. SpectralClustering

- `cluster(X,k,cfg)`
  - Suppose the input instance has a size of M. Initialize the similarity matrix S to be I_M
  - Calculate the distances of each pair of instances using `pdist()`, then transform the result back to MxM distance matrix using `squareform()`, denoted by dPair(i,j), the distance of $x^{(i)}$ and $x^{(j)}$
  - Calculate the similarity matrix according to input method
  - Compute the graph Laplacian matrix and the first k eigenvectors
  - Use the eigenvectors to do KmeansClustering
- ϵ-NN
  - For each instance, choose ϵ neighbors that are nearest to it according to the distance matrix excluding itself
  - If instance $x^{(t)}$ has a nearest neighbor $x^{(i)}$ then set S(t,i) and S(i,t) = dPair(t,i)



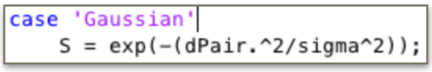
```
case 'eNN'
    [~,sortIndex] = sort(dPair, 2, 'ascend');
    minDist = sortIndex(:,1:epsilon+1);
    for i=1:size(X,1)
        S(i,minDist(i,2:epsilon+1)) = dPair(i,minDist(i,2:epsilon+1));
        S(minDist(i,2:epsilon+1),i) = dPair(minDist(i,2:epsilon+1),i);
    end
```

- ϵ-Ball
  - If dPair(i, j) < ϵ, then set S(i, j) = dPair(i, j)



```
case 'eBall'
    S(dPair<epsilon) = dPair(dPair<epsilon);
```

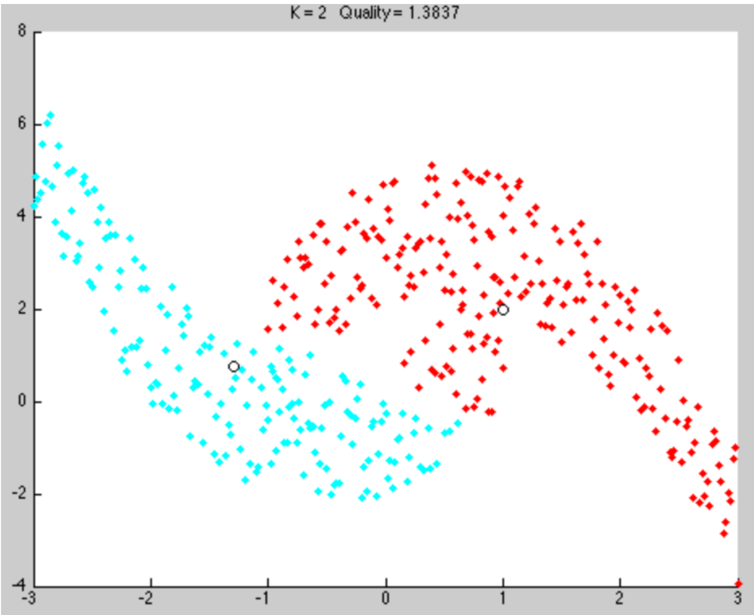
- Gaussian
  - Set S(i, j) = $\exp(-\frac{dPair(i,j)^2}{\sigma^2})$



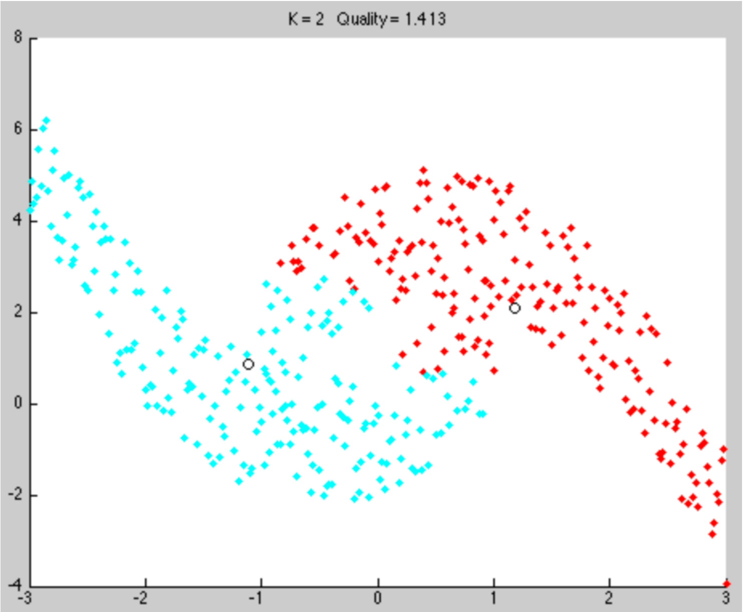
```
case 'Gaussian'
    S = exp(-(dPair.^2/sigma^2));
```

- Result of ϵ-NN :

ϵ=3



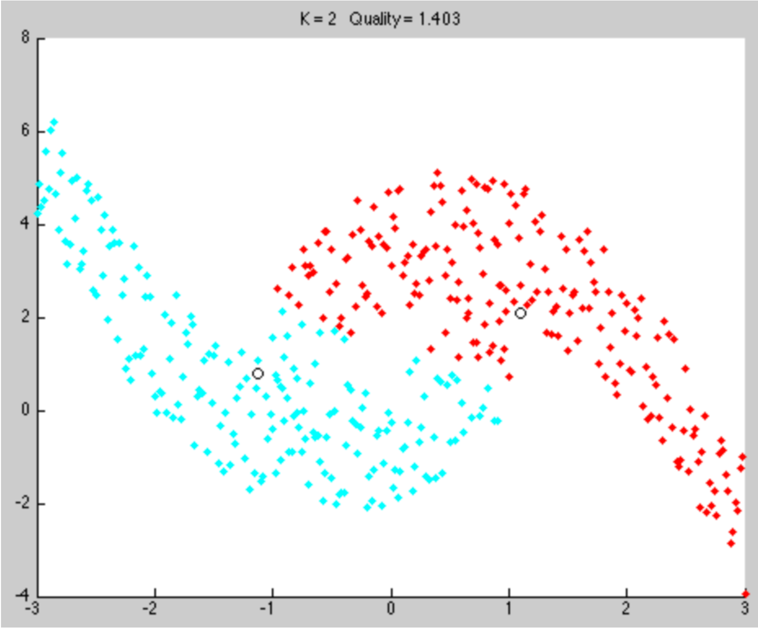
ϵ=10



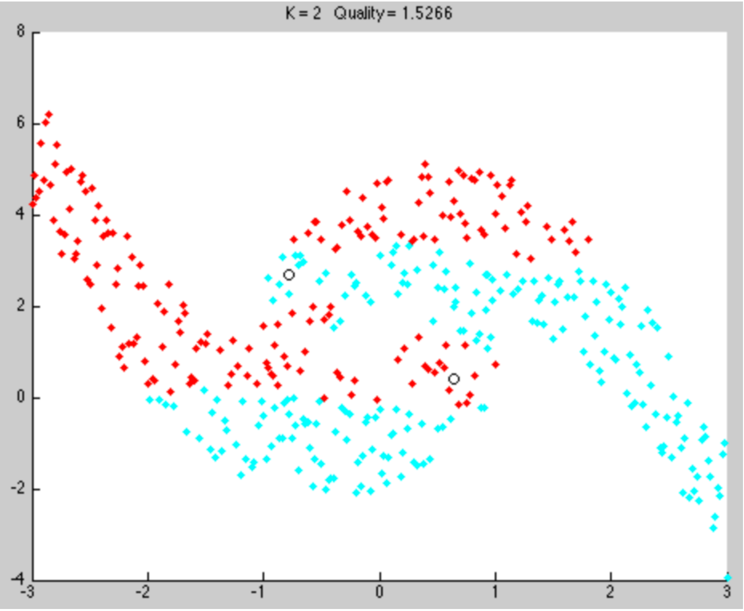
- The right result seems more correct because the size of clusters is more balanced

- Result of ϵ-Ball :

ϵ=1.2



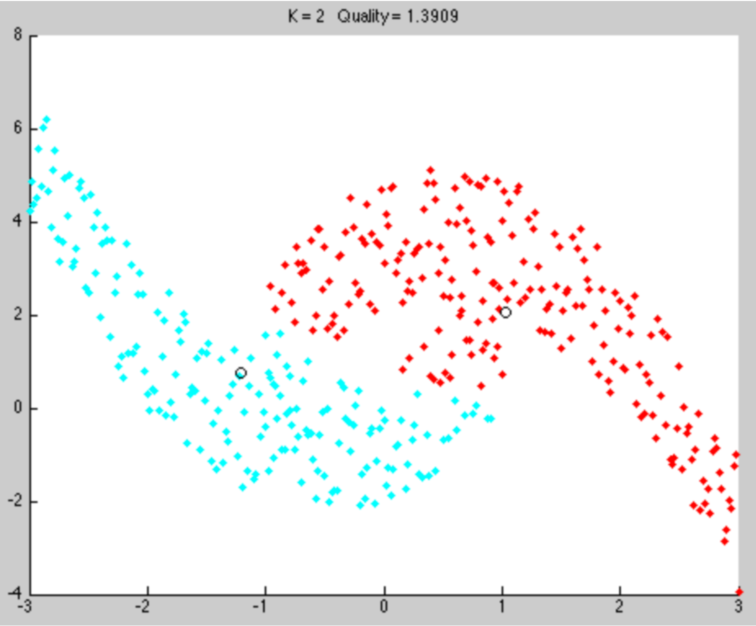
ϵ=5



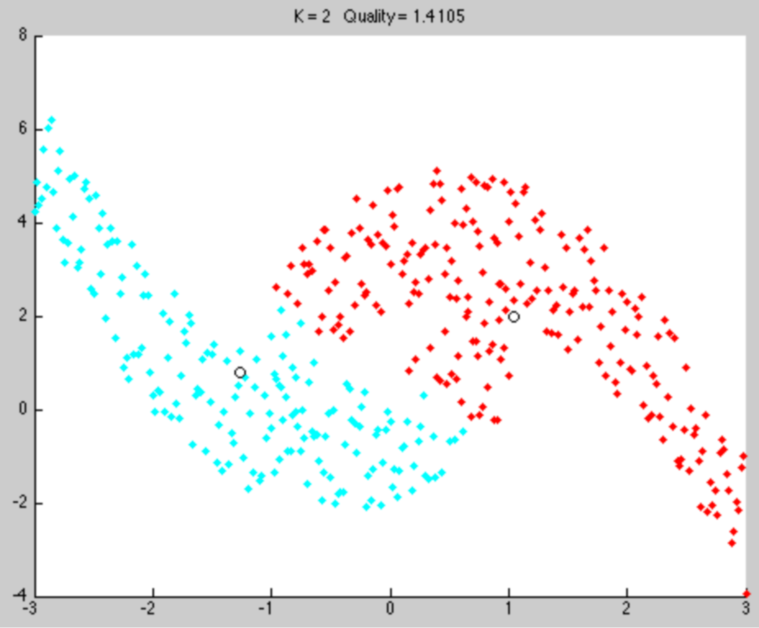
- Because the instances are close to each other, big ϵ leads to big error

- Gaussian :

σ=0.56



σ=0.93



- Change in σ does not severely affect the clustering result

3. Discussion of the quality metric

$$quality = \frac{\sum_{i,j}(m_i - m_j)^2}{\sum_i \frac{1}{|G_i|}\sum_{x \in G_i}(x - m_i)^2}$$

In KmeansClustering, when k=2, smaller the quality, better the result. But when k=6, the result is more reasonable for a greater quality.

In SpectralClustering with ϵ-NN similarity, greater quality leads to better result.

In SpectralClustering with ϵ-Ball similarity, smaller quality leads to better result.

In SpectralClustering with Gaussian similarity, quality does not necessarily reflect the correctness of result.

From the comparisons above, such quality metric is not an adequate measurement to evaluate the quality of clusters.