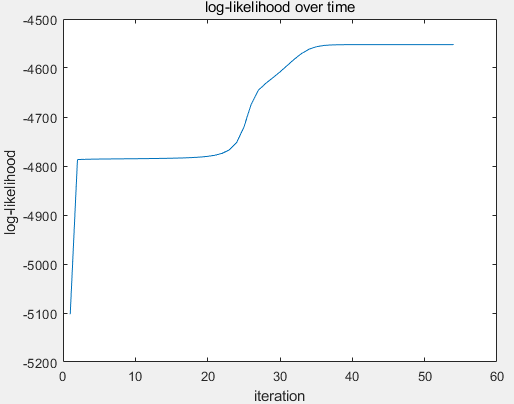
**ENGN2520 Homework 5**

**Ming Xu (Banner ID: B01532164)**

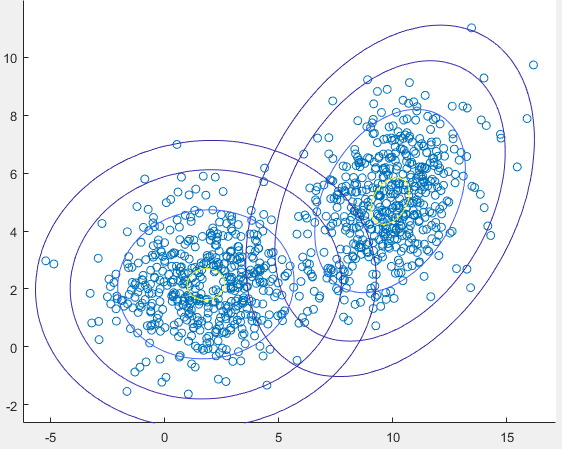
**Problem1--A mixture of 2 Gaussians**

***(a)*** ***A plot of the log-likelihood over time for the best choice of initial parameters***

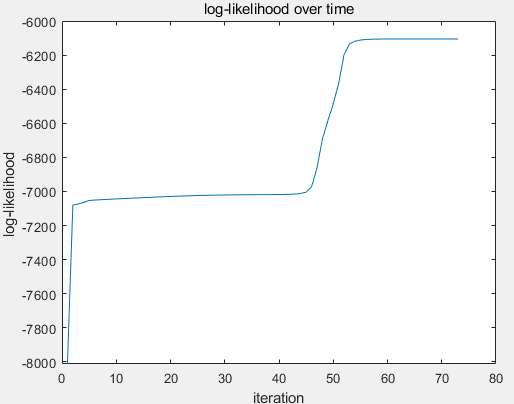


***(b)*** ***The resulting parameters of the mixture models.***

***(c)*** ***A visualization of the estimated means and covariances over the datasets.***

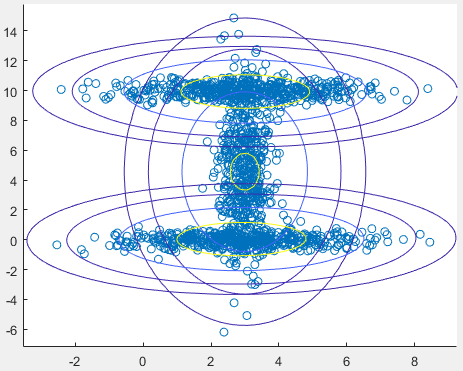


**Problem2 --A mixture of 3 Gaussians**

***(a)*** ***A plot of the log-likelihood over time for the best choice of initial parameters*** For multiclass 

***(b) The resulting parameters of the mixture models.***

***(c)*** ***A visualization of the estimated means and covariances over the datasets.***



**Source Code**

**1.The matlab function to calculate probability:**

function [prob] = probabilityCalculation(X,K,mu,sigma)

[N,~] = size(X);

prob = zeros(N, K);

for j = 1:K

prob(:, j) = mvnpdf(X, mu(j,:), squeeze(sigma(j,:,:)));

end

end

**2.The matlab function to implement ER algorithm:**

function [maxMU,maxSigma,maxPI,likelyhood] = EM\_Algorithm(data, K, numIterations , tolForInteration ,thresholdForCovMatrix)

%get dimension of input training data;

[N,D] = size(data);

%initialization

maxLikelyhood = -inf;

%iterate for certain times

for i = 1:numIterations

%set initial values for mu, pi, sigma of this iteration

%initialize mu by randomly selecting K data points

mu = data(randsample(N, K), :);

%initialize sigma by using the overall data covariance.

sigma = reshape(cov(data),1,D,D);

sigma = repmat(sigma, K, 1, 1);

pi = ones(1,K) / K;

%calculate the initial value of log-likelyhood

lastLikelyhood = -inf;

currentLikelyhood = sum(log(probabilityCalculation(data, K, mu , sigma) \* pi'));

allLikelyhoods = currentLikelyhood;

%loop until likelyhood becomes stable

while currentLikelyhood - lastLikelyhood > tolForInteration

lastLikelyhood = currentLikelyhood;

prob = probabilityCalculation( data, K, mu , sigma).\*pi;

%calculate rj and Nj

rj = prob./sum(prob,2);

Nj = sum(rj , 1);

%calculate mu

mu = (rj'\*data)./(Nj');

%calculate sigma

for j = 1:K

X\_mu = data-repmat(mu(j,:),N,1);

sigmaJ = (rj(:,j).\*X\_mu)'\*X\_mu;

sigmaJ = sigmaJ/Nj(j);

for d = 1:D

sigmaJ(d,d) = max(sigmaJ(d,d), thresholdForCovMatrix);

end

sigma(j, :, :) = sigmaJ;

end

%caculate pi

pi = Nj/N;

%update likelyhood

currentLikelyhood = sum(log(probabilityCalculation(data, K, mu, sigma) \* pi'));

%save current likelyhood

allLikelyhoods = [allLikelyhoods,currentLikelyhood];

end

%save the best likelyhood ever have

if currentLikelyhood>maxLikelyhood && currentLikelyhood<-1000

maxLikelyhood = currentLikelyhood ;

maxMU = mu;

maxSigma = sigma;

maxPI = pi;

likelyhood = allLikelyhoods;

end

end

end

**3. The matlab function for visualization:**

function [] = visualization(data,K,mu,sigma)

figure

hold on

%plot the training data

scatter(data(:,1), data(:,2))

maxValue = max(data, [], 1) + 1;

minValue = min(data, [], 1) - 1;

x1 = minValue(1) :.2: maxValue(1);

x2 = minValue(2) :.2: maxValue(2);

[X1,X2] = meshgrid(x1,x2);

X = [X1(:) X2(:)];

for k = 1:K

y = mvnpdf(X, mu(k,:), squeeze(sigma(k,:,:)));

y = reshape(y,length(x2),length(x1));

contour(x1 ,x2 ,y ,[0.0001 0.001 0.01 0.05 0.15 0.25 0.35]);

end

end

**4.Main Function:**

clc;clear;

numIterations = 20;

tolForInteration = 1e-7;

thresholdForCovMatrix = 1;

%% 2 Gaussian

load data2;

K = 2;

[mu, sigma,pi,likelyhood] = EM\_Algorithm(data, K, numIterations, tolForInteration, thresholdForCovMatrix);

pi

for i = 1:K

mu(i,:)

squeeze(sigma(i,:,:))

end

plot(likelyhood)

title('log-likelihood over time')

xlabel('iteration')

ylabel('log-likelihood ')

visualization(data, K, mu , sigma);

%% 3 Gaussian

load data3;

K = 3;

[mu, sigma,pi,likelyhood] = EM\_Algorithm(data, K, numIterations, tolForInteration, thresholdForCovMatrix);

pi

for i = 1:K

mu(i,:)

squeeze(sigma(i,:,:))

end

plot(likelyhood)

title('log-likelihood over time')

xlabel('iteration')

ylabel('log-likelihood ')

visualization(data, K, mu , sigma);