1. (a) # of iterations=225

Plot of objective function values versus iterations

|  |
| --- |
|  |

Compressed image compared to original image

|  |
| --- |
|  |

Difference image

|  |
| --- |
|  |

Nose region is less preserved because there are details in difference image while eyes and outer part of face is more preserved because there are nothing significant in difference image.

MATLAB script

|  |
| --- |
| close all; clear all; clc  load mandrill  k=1e2;    y=double(imread('mandrill.tiff'));  M=2;  n=prod(size(y))/(3\*M\*M);  d=size(y,1);  c=0;  x=zeros(n,3\*M\*M);  for i=1:M:d  for j=1:M:d  c=c+1;  x(c,:)=reshape(y(i:i+M-1,j:j+M-1,:),[1,M\*M\*3]);  end  end  dprime=size(x,2);  rng(0);  perm=randperm(n);  m=x(perm(1:k),:);    map\_old=zeros(n,1);  map=ones(n,1);    iter=1;  while norm(map-map\_old)>0  iter  map\_old=map;    dist=dist2(x,m);  [Min,map]=min(dist,[],2);    w(iter)=0;  for l=1:k  a=map-l;  zeroInd=find(a==0);  m(l,:)=mean(x(zeroInd,:));  w(iter)=w(iter)+sum(sum((x(zeroInd,:)-repmat(m(l,:),[size(x(zeroInd,:),1) 1])).^2,2));  end  iter=iter+1;  end  for l=1:k  a=map-l;  zeroInd=find(a==0);  x1(zeroInd,:)=repmat(m(l,:),[numel(zeroInd) 1]);  end  c=0;  for i=1:M:d  for j=1:M:d  c=c+1;  y1(i:i+M-1,j:j+M-1,:)=reshape(x1(c,:),[M M 3]);  end  end  figure  plot(w)  xlabel('Iterations');  ylabel('W(c)')    figure;  subplot(121)  imagesc(y1/256)  title('Compressed')  subplot(122)  imagesc(y/256)  title('Original')  figure;  dy=y1-y;  imagesc(dy); |

(b)

24\*M^2\*k bits for cluster

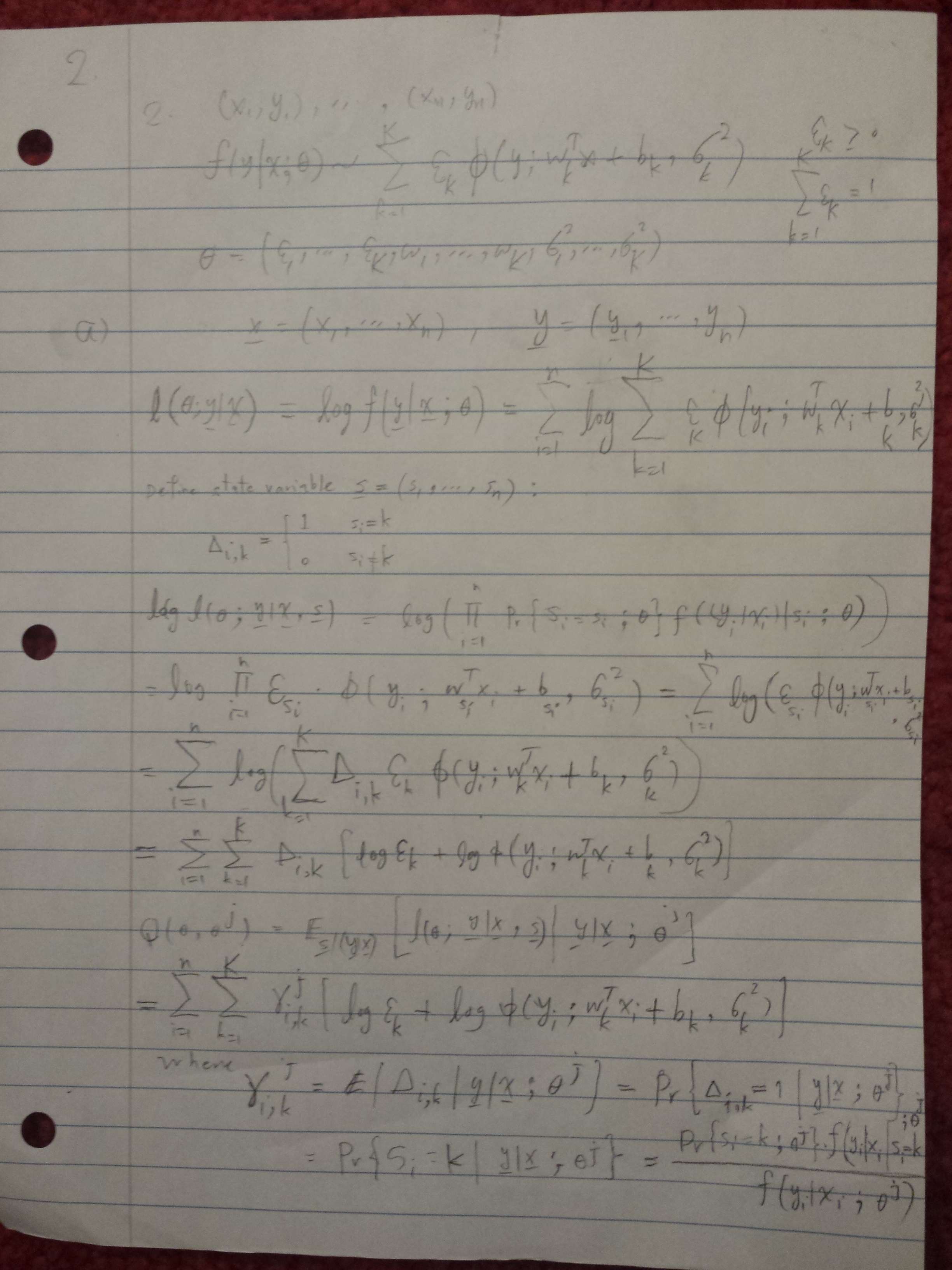
(N/M)^2 blocks & for each bock log2k bits for cluster number

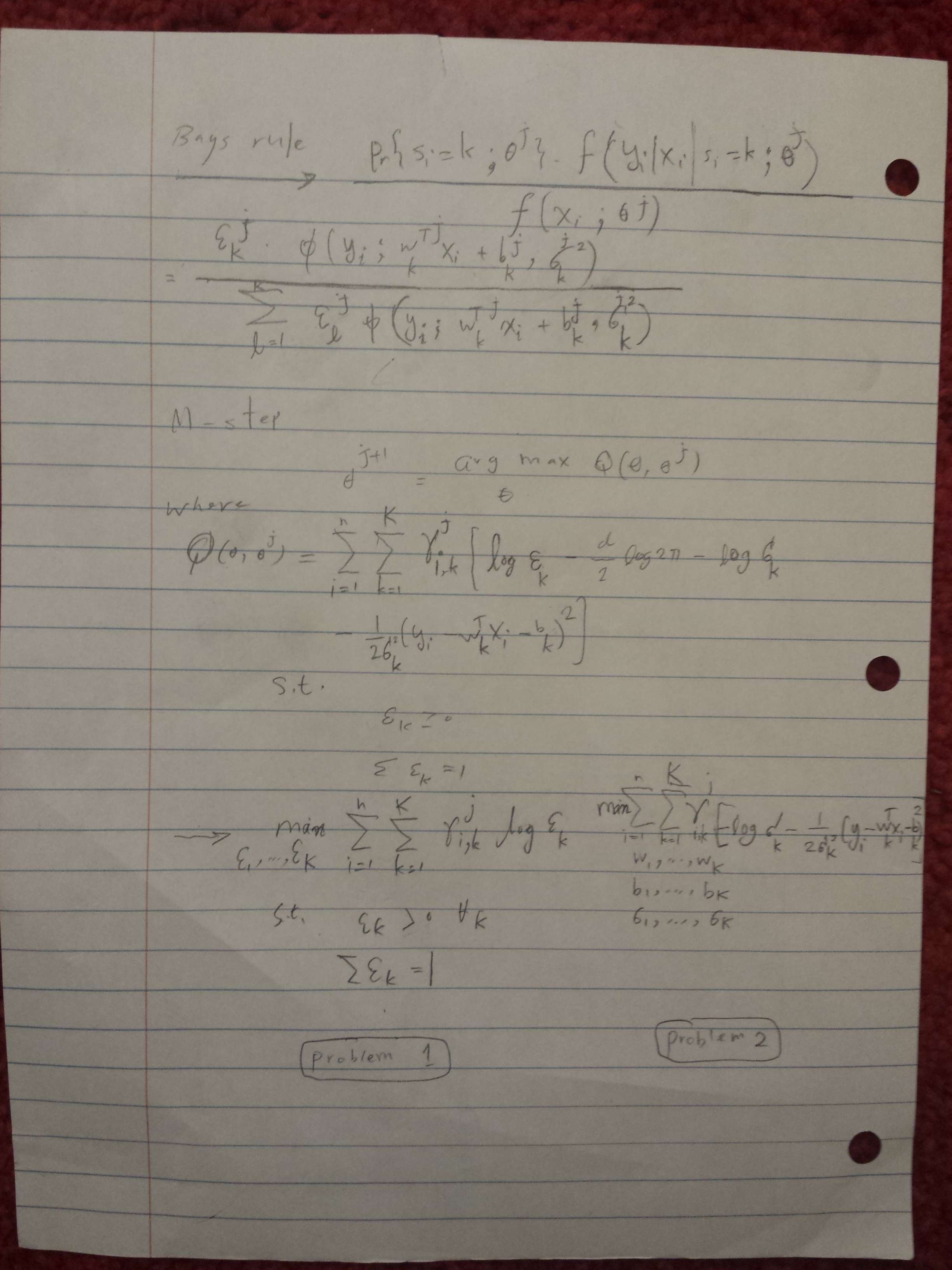
Bpp of compressed=(24\*M^2\*k+(N/M)^2\* log2k)/(24\*N^2)

Compression ratio= 0.0744

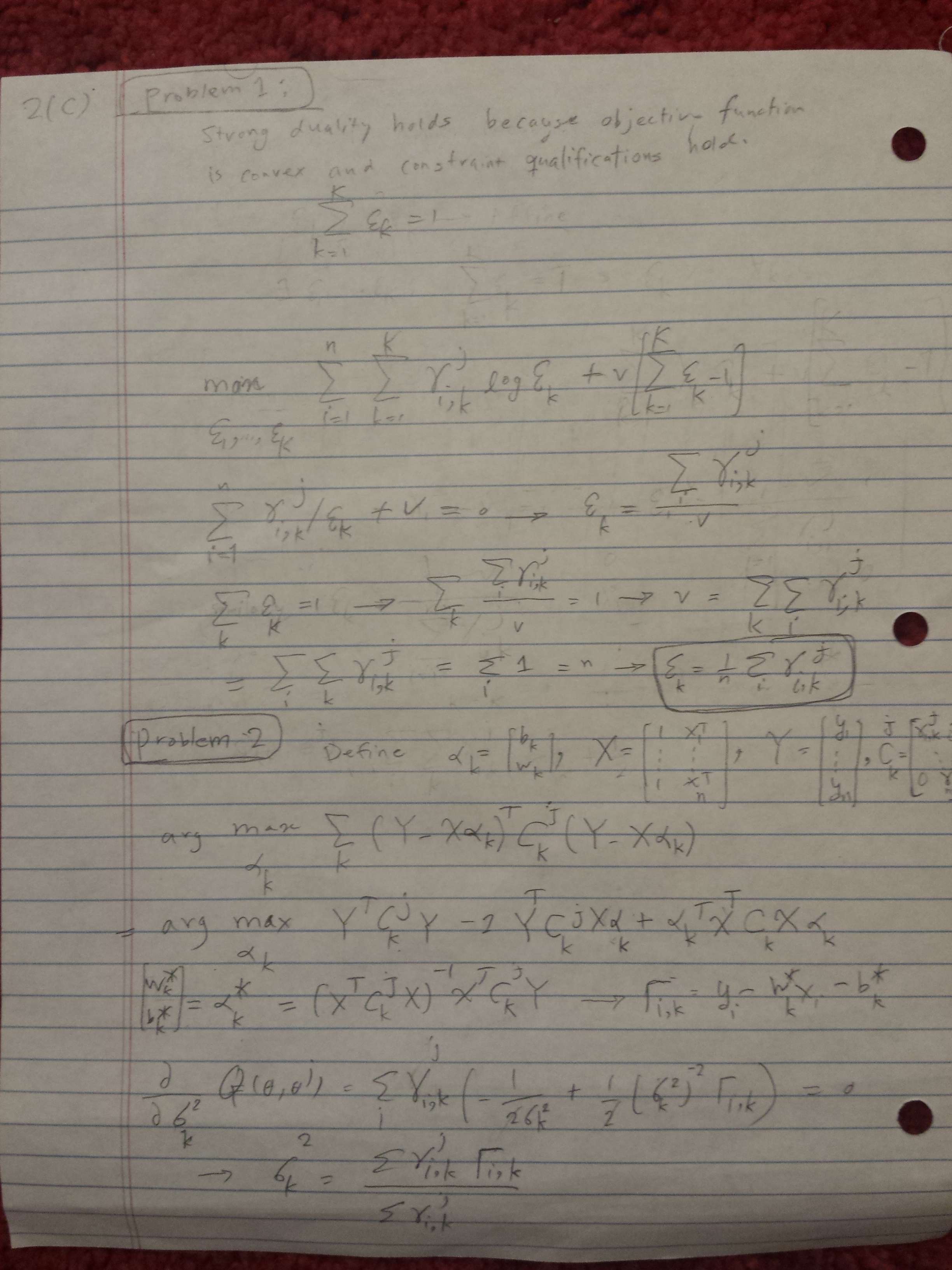
Relative mean absolute error=0.002

2(a)





2(c)



2(d)

# of iterations=51,

|  |  |
| --- | --- |
|  |  |

ϵ=[0.2062,0.7938], w=[1.2834,-1.9516], b=[-0.6777,0.5280], ϭ^2=[0.0747,0.2194]

MATLAB script

|  |
| --- |
| close all; clear all; clc  rng(0);  n = 200;  K = 2;  e = [.7 .3];  w = [-2 1];  b = [.5 -.5];  v = [.2 .1];  for i=1:n  x(i) = rand;  if rand < e(1);  y(i) = w(1)\*x(i) + b(1) + randn\*sqrt(v(1));  else  y(i) = w(2)\*x(i) + b(2) + randn\*sqrt(v(2));  end  end  figure  plot(x,y,'bo')  hold on  t=0:0.01:1;  plot(t,w(1)\*t+b(1),'k')  plot(t,w(2)\*t+b(2),'k')  ek = [.5 .5];  wk = [1 -1];  bk = [0 0];  sk = repmat(var(y),1,2);  etak = zeros(2,K);  rik =zeros(n,K);  cont = true;  stopThreshold = 10 ^ (-4);  stepCount = 0;  pk = zeros(n,K);  phi = zeros(n,1);  X = [ones(n,1) x(:)];  thetaik = zeros(n,K);  theta = 0;  thetaOld = 0;  thetaHistory = zeros(10000,1);  L = 0;  LOld = 0;  Li = zeros(n,1);  LHistory = zeros(10000,1);  firstFlag = 1;  while cont == true  stepCount = stepCount + 1;    % E-step  for sIndex = 1 : n  for kIndex = 1 : K  gPdf = normpdf(y(sIndex), wk(kIndex) \* x(sIndex) + bk(kIndex),...  sqrt(sk(kIndex)));  rik(sIndex, kIndex) = ek(kIndex) \* gPdf;  end  if(sum(rik(sIndex,:)) ~= 0)  rik(sIndex,:) = rik(sIndex,:) / sum(rik(sIndex,:));  else  keyboard  end  end  % M-step  for kIndex = 1 : K  ek(kIndex) = sum(rik(:,kIndex))/n;  ck = diag(rik(:,kIndex));  etak(:,kIndex) = (X' \* ck \* X) \ X' \* ck \* y';  bk(kIndex) = etak(1, kIndex);  wk(kIndex) = etak(2, kIndex);  for sIndex = 1 : n  pk(sIndex,kIndex) = (y(sIndex) - wk(kIndex) \* x(sIndex) - bk(kIndex)) ^ 2;  end  sk(kIndex) = sum(rik(:,kIndex) .\* pk(:,kIndex)) / sum(rik(:,kIndex));  end  Li = zeros(n,1);  for sIndex = 1 : n  for kIndex = 1 : K  Li(sIndex, 1) =Li(sIndex, 1) + ek(kIndex) \* normpdf(y(sIndex),wk(kIndex) \*...  x(sIndex) + bk(kIndex), sqrt(sk(kIndex)));  end  end  L = sum(log(Li(:,1)));  if abs(LOld - L) < stopThreshold  if firstFlag == 1  LOld = L;  LHistory(stepCount) = L;  firstFlag =0;  else  LHistory(stepCount) = L;  cont = false;  end    else  LOld = L;  LHistory(stepCount) = L;  firstFlag =0;  end  end  plot(t,wk(1)\*t+bk(1),'r--')  plot(t,wk(2)\*t+bk(2),'r--')  figure  plot(LHistory(1:stepCount))  xlabel('Iterations');  ylabel('Log liklihood') |

