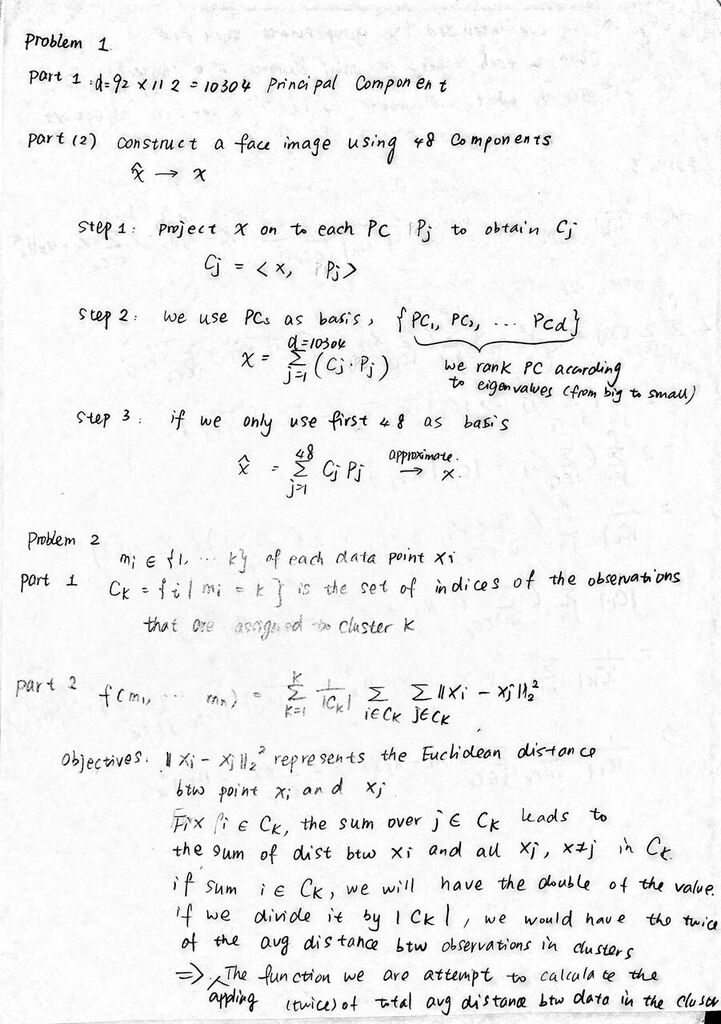
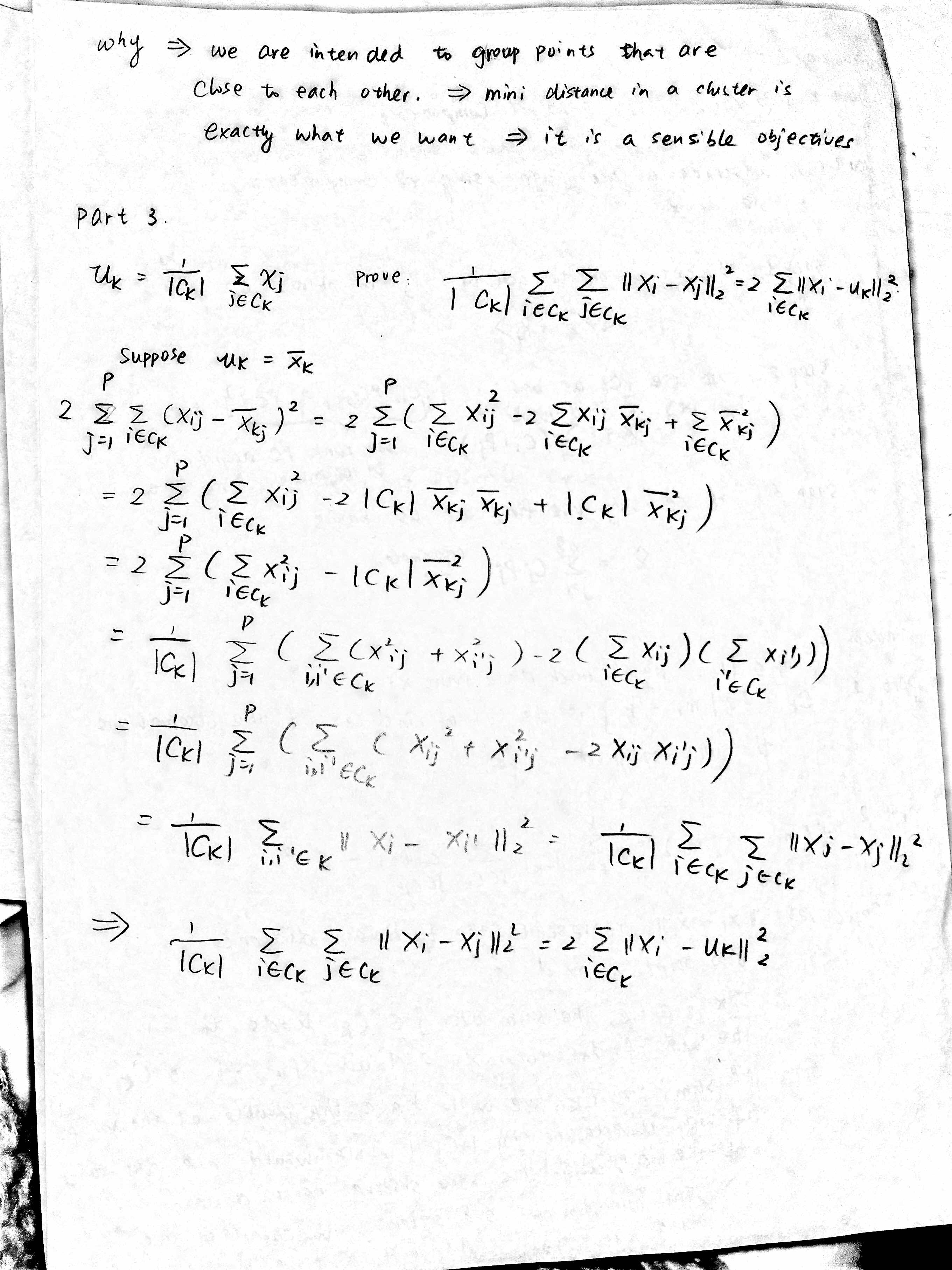
HW4

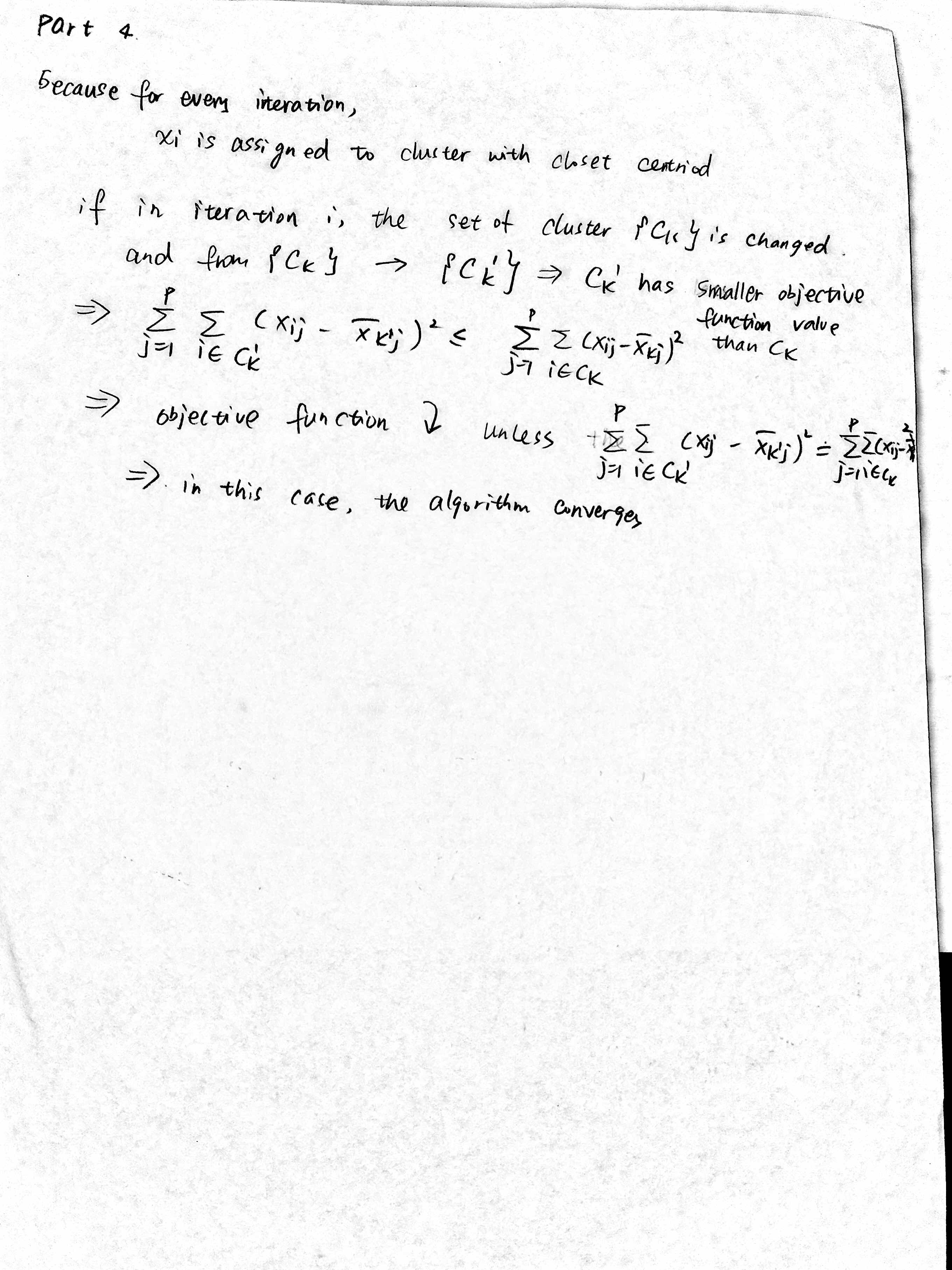
STAT 4400

Name: Jiahong Hu

UNI:jh3561







Problem 3

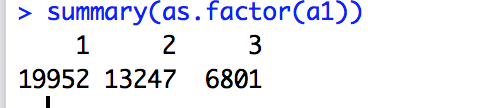
Part 1

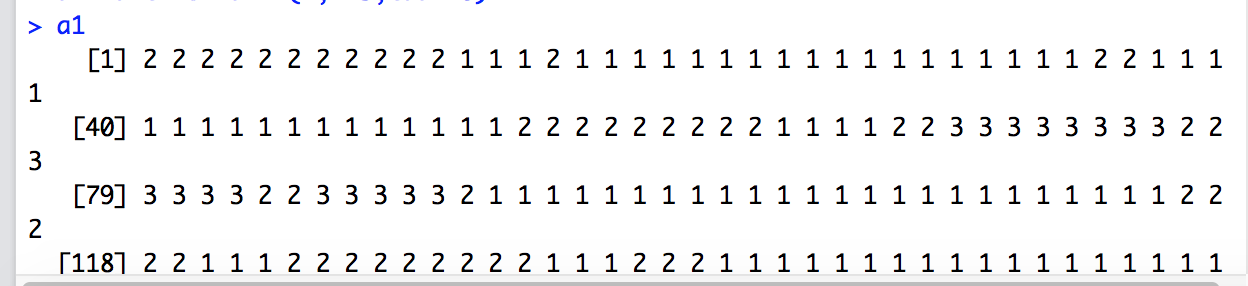
See Code In the End

Part 2

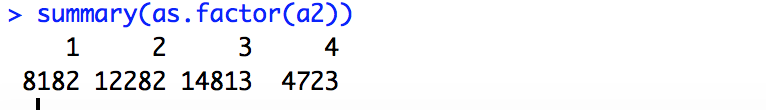
I choose tau=10,50,100. I think tau=10,100 give a reasonable result and tau =100 gives too much tolerance.

* When tau=10, K=3

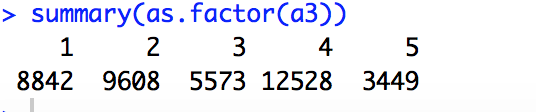




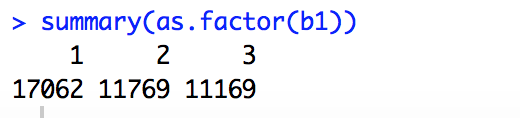
* when tau=10, K=4



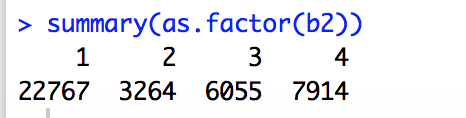
* when tau=10, K

-

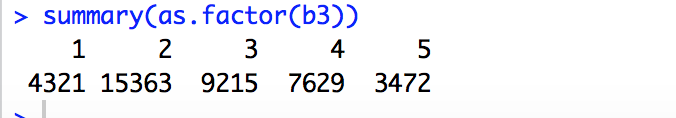
* When tau=50, K=3



* When tau =50, K=4



* When tau= 50, K=5

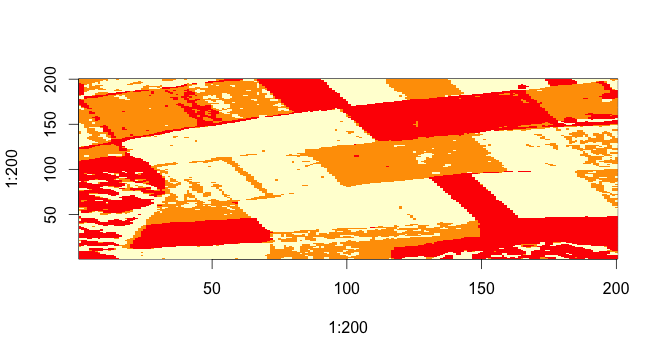


I figure out that there are some difference between tau=10 and tau=50.

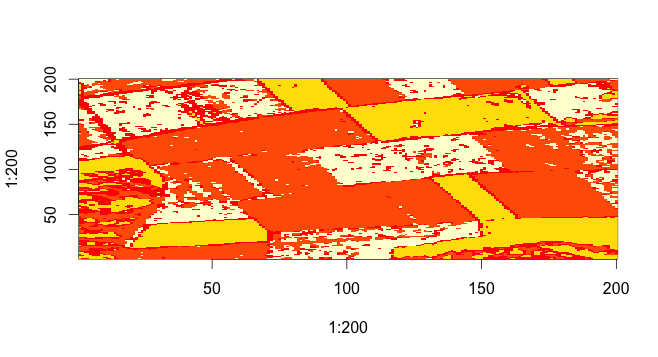
I will choose tau =10 for further demonstration

Part c

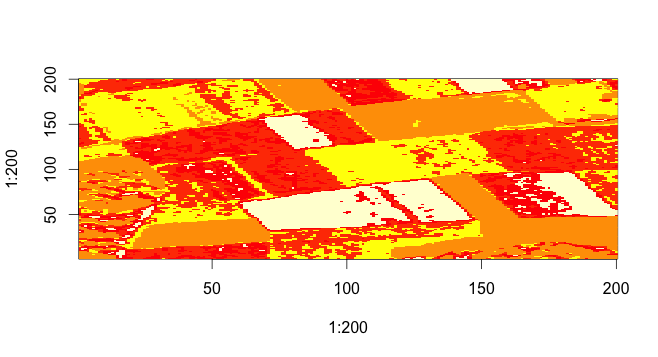
Tau=10, K=3



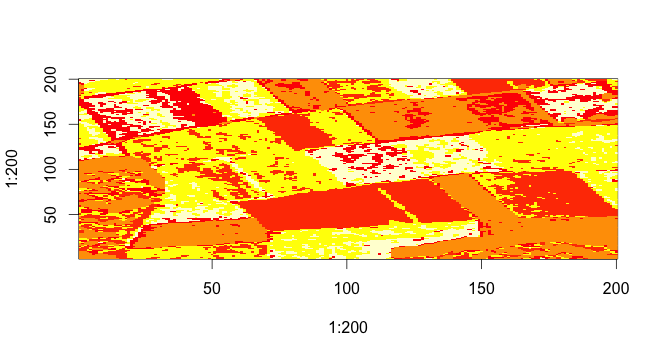
Tau=10, K=4



Tau=10, K=5



Tau=50, K=5



Code

setwd("/Users/jiahonghu/Desktop/hw4")

H<-matrix(readBin("histograms.bin","double",640000),40000,16)

dim(H)

for(i in 1:40000){

for(j in 1:16){

if (H[i,j]==0){

H[i,j]=H[i,j]+0.01

}

}

}

##Problem 1##

#EM algorithm

###input###

#H - dim=n\*d

#K - number of clusters

#tau - terminate threshold paramter

MultinomialEM=function(H,K,tau){

n=nrow(H) # number of histograms

d=ncol(H) # number of bin per histogram

delta=1000 # meansure the change of assigments between iteraiton i and i+1

i=1 # the number of iteration of the algorithm

prop=rep(1/K,K) # initiate a cluster proportion

# step 1:Initial centriod by selecting K rows from n rows of H matrix

# ctd is a matrix of dim = k\*d and represents the centriod of K clusters, each row represents a centriod of a cluster

set.seed=1

ctd=H[sample(1:n,K),]

dim(ctd)

# step 2: EM algorithm

# the algorithm stops only if the change is less than tau

while(delta>tau){

# E-step

phi=exp(H%\*%t(log(ctd)))

phi\_cp=phi\*prop

row\_sum\_phi\_cp=apply(phi\_cp,1,sum)

inv\_row\_sum=1/row\_sum\_phi\_cp

A\_matrix=matrix(rep(inv\_row\_sum,K),n,K)

A=A\_matrix\*phi\_cp

# M-step

ctd\_current=ctd # theta at the iteration i

ctd=t(A)%\*%H

col\_sum\_ctd=apply(ctd,1,sum)

inv\_col\_sum=1/col\_sum\_ctd

ctd\_matrix=matrix(rep(inv\_col\_sum,d),K,d)

ctd=ctd\_matrix\*ctd # new theta at the iteraiton i +1

prop\_sum=sum(apply(A,2,sum))

prop=prop/prop\_sum # update proportion for clusters

i=i+1

delta=norm(ctd\_current-ctd,type="O")

}

return(result=apply(A,1,which.max))

}

##Problem 2 ##

# for tau=10,50,100

a1=MultinomialEM(H,K=3,tau=10)

a1

summary(as.factor(a1))

a2=MultinomialEM(H,K=4,tau=10)

a2

summary(as.factor(a2))

a3=MultinomialEM(H,K=5,tau=10)

a3

summary(as.factor(a3))

b1=MultinomialEM(H,K=3,tau=50)

summary(as.factor(b1))

b2=MultinomialEM(H,K=4,tau=50)

summary(as.factor(b2))

b3=MultinomialEM(H,K=5,tau=50)

summary(as.factor(b3))

## problem 3 ##

img\_function=function(x){

x\_matrix=matrix(x,nrow=200,byrow=TRUE)

img\_matrix=NULL

ind=nrow(x\_matrix)

for( i in 0:(ind-1)){

img\_matrix=cbind(img\_matrix,x\_matrix[ind-i,])

}

image(x=1:200,y=1:200,img\_matrix)

}

img\_function(a1)

img\_function(a2)

img\_function(a3)

img\_function(b3)