SURE Project and Weekly Write-up

Yunqiu Guo

Summer 2018

1 Week One (June 25/27/29)

1.1 Task One

Create several test data.

1. small clusters: data points are generated deterministically(?)
   Testdata 1: generate two $1 * 6$ vectors of normally distributed numbers
2. small random clusters: (in terms of one-dimensional)
   Test example given in class

1.2 Task Two

1. Generate the corresponding Linear Program for the small data set.
2. Solve the LP (Matlab or CVX)
3. Verify if the result is integral
4. Find the corresponding clusters.
5. Find the corresponding feasible dual solution

1.3 Toy Example

Clustering is a task of grouping a set of objects in such a way that objects in the same group or same cluster are much more similar to each other compared to objects in other clusters. Let us first consider the following simple clustering example.

Consider the small clustering example:

Data points set given is:

$$X = [x_1, x_2, x_3, x_4, x_5, x_6]$$

which is six points on the x-axis, and corresponding value is

$$-11, -10, -9, 9, 10, 11$$

We need to group the set X into two clusters and find the center for each group. This is already an easy clustering problem.

We can tell easily that x_1, x_2, x_3 form one cluster, and x_4, x_5, x_6 form another cluster.

How to represent all of the information in terms of linear program? And get the corresponding integral solution?

First of all, integer programming problem is

$$\begin{array}{lll} \max & c^T x & \\ s.t. & Ax & \leq \quad b \\ & x & \geq \quad 0 \\ and & & x \in \mathbb{Z}^n \end{array}$$

If we add the slack variable s to it, we can get the standard integer LP

$$\begin{array}{lllll} \max & c^T x & & & \\ s.t. & Ax & + \quad s & = & b \\ & s & \geq \quad 0 & & \\ & x & \geq \quad 0 & & \\ and & x \in \mathbb{Z}^{\ltimes} & & & \end{array}$$

When we are writing the LP for a clustering problem (say k-median clustering problem), we are actually doing an integer programming problem. Since if we have two indicative variable

$$z_{pq}: \begin{array}{ll} 1 & \textit{if the point } p \textit{ is the center for point } q \\ 0 & \textit{otherwise} \end{array}$$

$$y_p: \begin{array}{ll} 1 & \textit{if the point } p \textit{ is the center for one of the cluster} \\ 0 & \textit{otherwise} \end{array}$$

Then our goal is to solve for z_{pq} and y_p for the objective function to minimize the sum of distance between all the points in the data set.

$$\begin{array}{lrcl} \min_{z \in \mathbb{R}^{\mathbb{N} \times \mathbb{N}}} & \sum_{p,q \in P} d(p,q) z_{pq} & & \\ s.t. & \sum_{p \in P} z_{pq} & = & 1 \\ & z_{pq} & \leq & y_p \\ & \sum_{p \in P} y_p & = & k \\ & z_{pq} & \in & \{0,1\} \\ & y_p & \in & \{0,1\} \end{array}$$

And for solving this integer programming, we need to relax the constraints from integer $\{0, 1\}$ to the interval $[0, 1]$

Then we can get the relaxation for k-median clustering problem

$$\begin{array}{lrcl} \min_{z \in \mathbb{R}^{\mathbb{N} \times \mathbb{N}}} & \sum_{p,q \in P} d(p,q) z_{pq} & & \\ s.t. & \sum_{p \in P} z_{pq} & = & 1 \\ & z_{pq} & \leq & y_p \\ & \sum_{p \in P} y_p & = & k \\ & z_{pq} & \in & [0,1] \\ & y_p & \in & [0,1] \end{array}$$

Now, we can try to write the corresponding LP for our simple example.

The z_{pq} vector is $\mathbb{R}^{1\times 36}$

$$z_{pq} = [z_{11}z_{12}z_{13}...z_{22}...z_{64}z_{65}z_{66}]$$

The y_p vector is $\mathbb{R}^{1\times 6}$

$$y_p = [y_1y_2y_3y_4y_5y_6]$$

In the objective function the c vector is the vector of distance between each pair of points.

Our main goal is to write out each constraints explicitly and find the corresponding A, b, c for our LP.

```
%vector c
c =[0 1 2 20 21 22 1 0 1 19 20 21 2 1 0
18 19 20 20 19 18 0 1 2 21 20 19
1 0 1 22 21 20 2 1 0 zeros(1,6)] ;
% c is a 1*36 vector
b1 = ones(6,1);
b2 = zeros(36,1);
beq = [b1;2];
I = eye(6);
Y = zeros(6,6);
l1 = zeros(1,36);
l2 = ones(1,6);
l12 = [l1,l2];
Aeq = [repmat(I,1,6),Y;l12];
b= b2;
I1 = eye(36);
y = zeros(6,1);
YY = kron (I,-b1);
A = [I1 YY];
%lower bound and upper bound
lb = zeros(1,42);
ub = ones(1,42);
zy = linprog(c,A,b,Aeq,beq,lb,ub)

Optimal solution found.


zy =

       0
       0
       0
       0
       0
       0
       1
       1
       1
```

0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
1
1
1
0
0
0
0
0
0
0
0
1
0
0
1
0

We can see from the result that z_{21}, z_{22}, z_{23} and z_{51}, z_{52}, z_{53} are 1, which means that x_2, x_5 are the two centers of three points. and y_2, y_5 is also equal to 1, indicates that x_2, x_5 are the centers.

1.4 Generalized Input Data sets

For more complicated data sets, we can write a function to give out the corresponding $k-$ median LP with the given X data sets and the number of clusters k. That is to say, we can write a function to calculate the corresponding A, b, c vector for the LP.

```
function [A,b,c,Aeq,beq]= lin(x,k)
    n = length(x);
    I = eye(n);
```

```
    beq = [ones(n,1);k];
    Y = zeros(n,n);
    l1 = zeros(1,n*n);
    l2 = ones(1,n);
    l12 = [l1 l2];
    Aeq = [repmat(I,1,n),Y;l12];
    b = zeros(n*n,1);
    I1 = eye(n*n);
    b1 = ones(n,1);
    I2 = eye(n);
    YY = kron(I2,-b1);
    YY
    A = [I1 YY];
    %use norm to construct the c vector
    k = 1;
    for i = 1:n
        for j = 1:n
            d(i,j) = norm (x(:,j)-x(:,i));
            c(k) = d(i,j);
            k = k + 1;
        end
    end
    c = [c zeros(1,n)];
    A
    b
    c
    Aeq
    beq
end
```

2 Week Two (July 2/ July 5)

In the paper *"Relax, No need to round - Integrality of Clustering Formulations"*, the authors study on a specific distribution over data, which is the data points drawn i.i.d. from disjoint balls of equal radius, and each ball contains equal number of points n.

Now here, it's interesting to investigate over:

1. data points drawn i.i.d. from disjoint balls of different radius, and each ball contains equal number of n points.

2. data points drawn i.i.d. from disjoint balls of same radius, but each ball contains different number of n_i points.

3. Balls overlap with each other (or the points are drawn according to the mixture of Gaussians).

   For the GMM, there's no "ground truth" clustering to recover, therefore it's not even clear how to build a dual certificate to certify an integral solution.

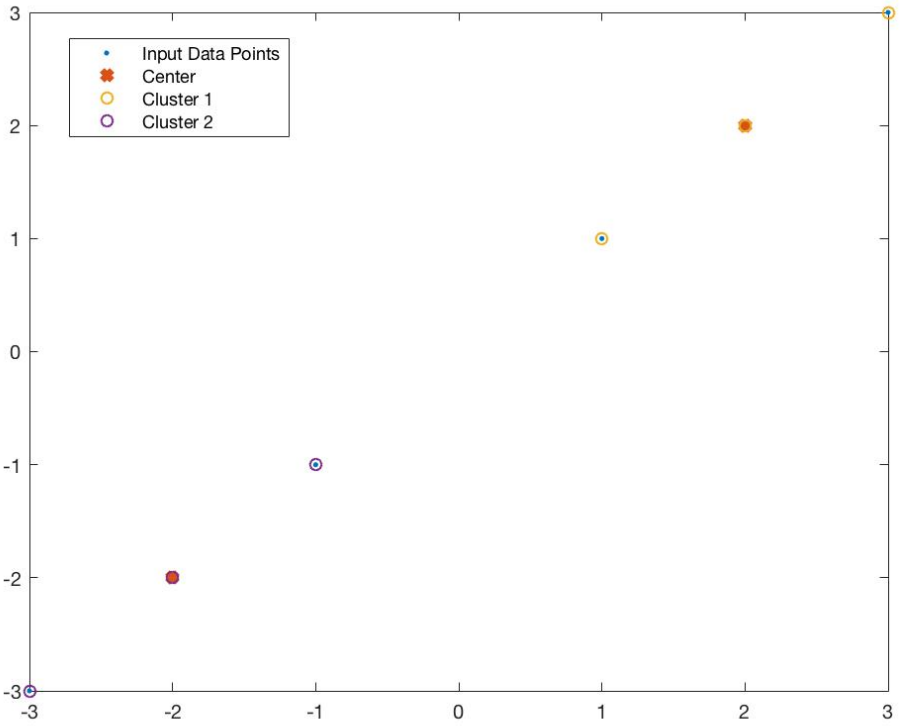
As also mentioned in the paper, for k-median LP, there's a high probability to get the integral solution. Even if the failure instances still coincide with clusterings, just not the planted disjoint supports.

2.1 Computational results and examples

2.1.1 Toy Examples -Six points $\in \mathbf{R}^2$

The input data points are the following six points $(1,1), (2,2), (3,3), (4,4), (5,5), (6,6)$

Num of Clusters k	Num of points in each cluster n	Integrality?	Recover True Cluster?
2	3	Yes	Yes




```
x = [1 2 3 -1 -2 -3; 1 2 3 -1 -2 -3];
k = 2;
title('Toy Six Points');
plot(x(1,:),x(2,:),'.');
pause;
hold on
[A,b,c,Aeq,beq]= lin(x,k);
lb = zeros(1,42);
ub = ones(1,42);
result = linprog(c,A,b,Aeq,beq,lb,ub);
zpq = result(1:36,:);
```

```
zz = reshape(zpq,[6,6]);
zz = zz';
yp = result(37:42,:);
ctr = find(yp);
ctr
c1 = x(:,ctr(1));
c2 = x(:,ctr(2));
cc = [c1 c2];
cc
plot(cc(1,:),cc(2,:),'x','LineWidth',5);
pause;
hold on
gp1 = zz(ctr(1),:);
g1 = find(gp1);
gg1 = x(:,g1);
plot(gg1(1,:),gg1(2,:),'o');
pause;
gp2 = zz(ctr(2),:);
g2 = find(gp2);
gg2 = x(:,g2);
plot(gg2(1,:),gg2(2,:),'o');
legend('Input Data Points','Center','Cluster 1','Cluster 2');
pause;
gp = [gp1;gp2];
```

Numerical results:

```
>> toysixpoints

Optimal solution found.


zpq =

     0
     0
     0
     0
     0
     0
     1
     1
     1
     0
     0
     0
     0
     0
     0
```

0
0
0
0
0
0
0
0
0
0
0
0
1
1
1
0
0
0
0
0
0

zz =

0	0	0	0	0	0
1	1	1	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	1	1	1
0	0	0	0	0	0

yp =

0
1
0
0
1
0

ctr =

2
5

cc =

$$\begin{matrix} 2 & -2 \\ 2 & -2 \end{matrix}$$

2.1.2 Two disjoint balls with same radius and same simulated number of points.

As also used in the paper, if the given data points are drawn from two separated disjoint balls. The result is highly integral and also the clustering result recovers the true cluster.

Num of Clusters k	Num of points in each cluster n
2	10

Centers	Radius	Integrality	Recover True Cluster?
(1,1) and (5,5)	1	Yes	Yes

Fig 2.1.2 Simulated 20 data points from two separated disjoint balls with same radius.

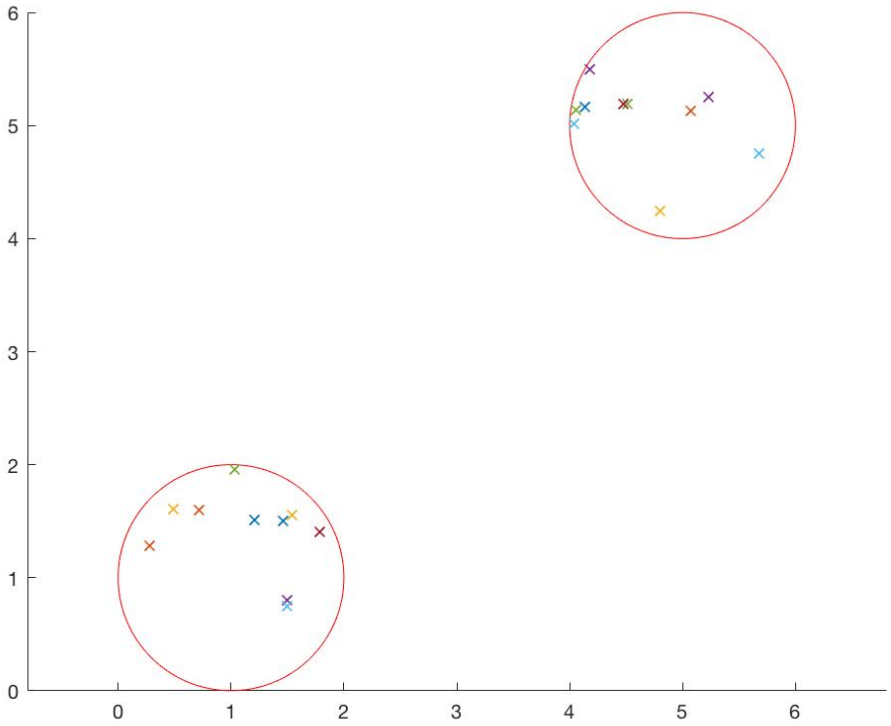


Fig 2.1.2 Found two centers by linprog.

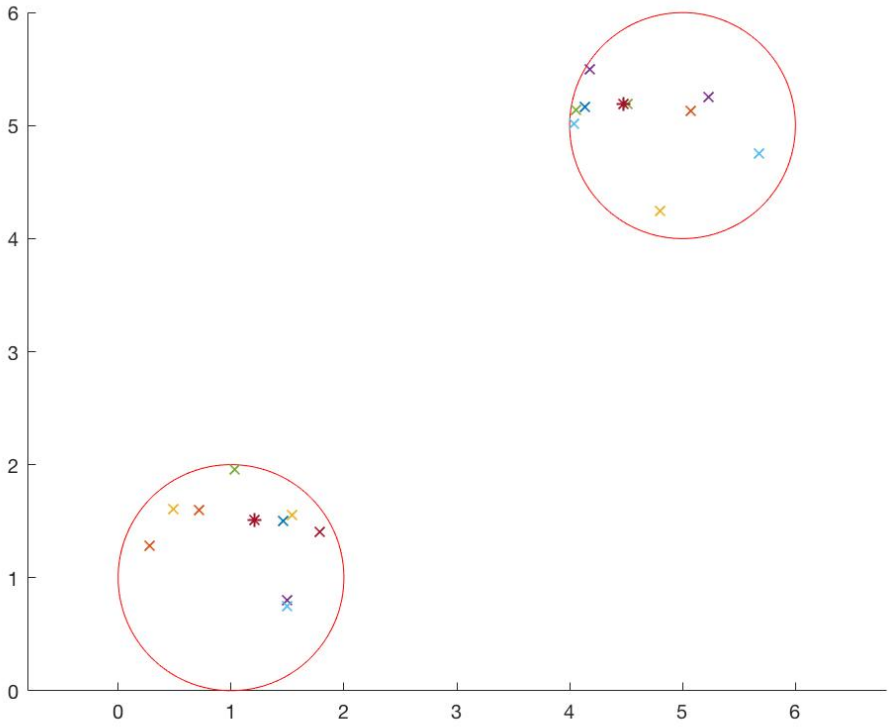


Fig 2.1.2 Cluster 1

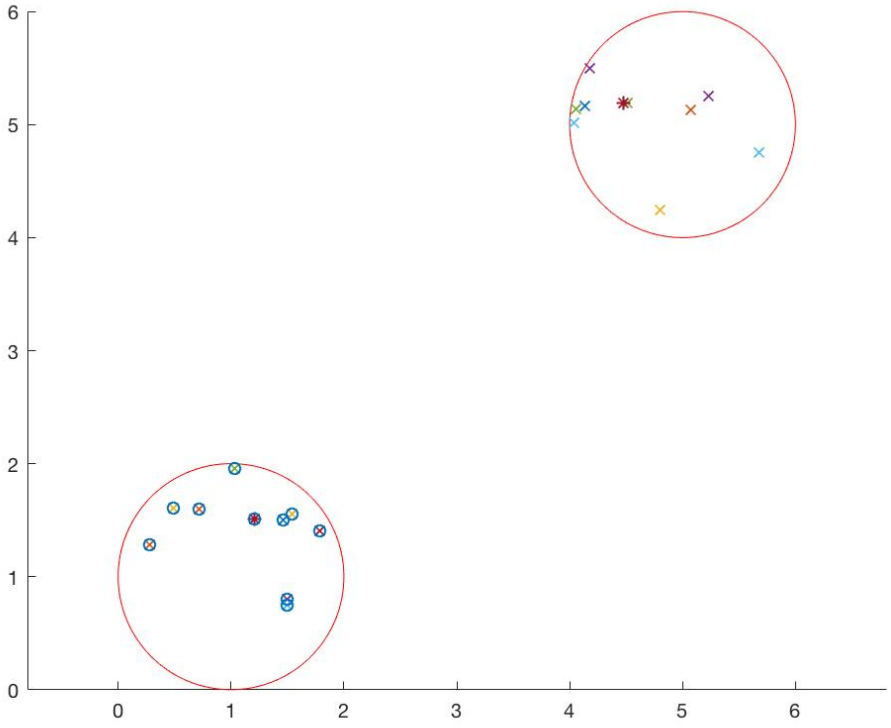
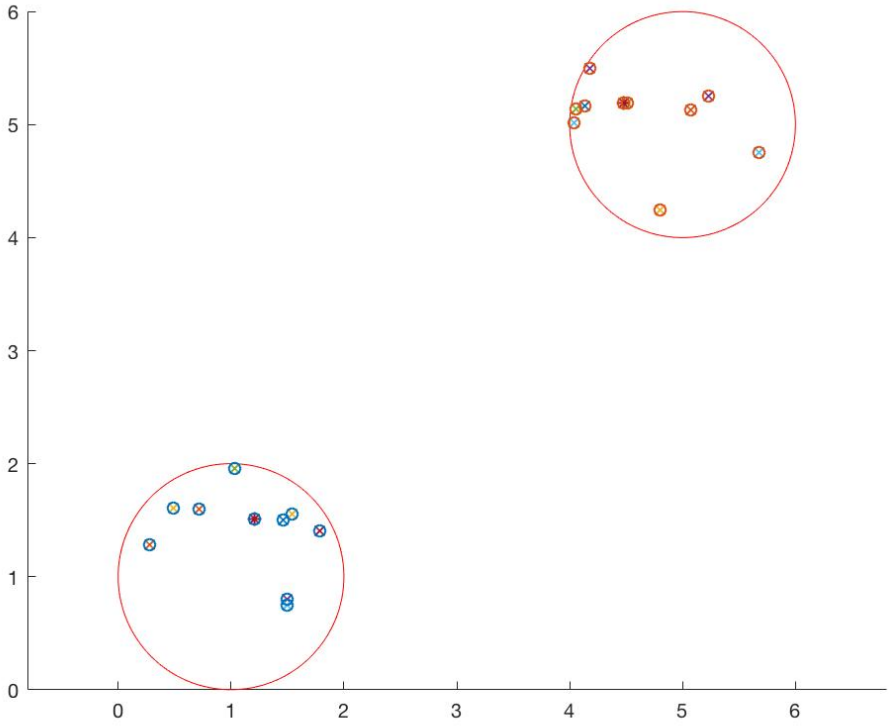
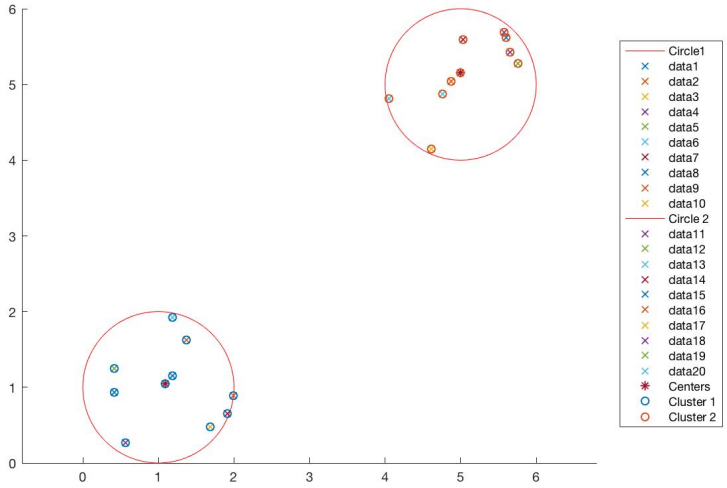


Fig 2.1.2 Cluster 2



Do the experiment again, the one-step result will look like the following plot.



(The simulated twenty points are different points.)

```
function [x y]=cirrdnPJ(x1,y1,rc)
    a=2*pi*rand;
    r=sqrt(rand);
    x=(rc*r)*cos(a)+x1;
    y=(rc*r)*sin(a)+y1;
end
```

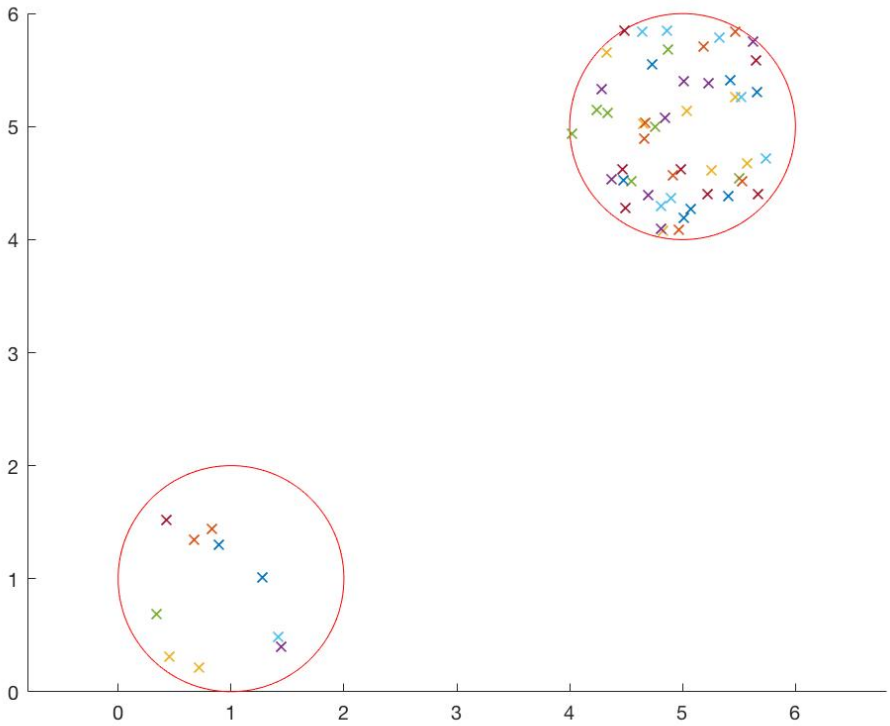
```
clf
axis equal
hold on
x1=1;
y1=1;
rc1=1;
[x,y,z] = cylinder(rc1,200);
plot(x(1,:)+x1,y(1,:)+y1,'r')

X1 = zeros(10,2);
for t=1:10
[x,y]=cirrdnPJ(x1,y1,rc1);
 X1(t,:) = [x y];
plot(x,y,'x');
end
hold on
x2= 5;
y2 = 5;
rc2 = 1;
[x,y,z] = cylinder(rc2,200);
plot(x(1,:)+x2,y(1,:)+y2,'r')
X2 = zeros(10,2);
for t=1:10
[x,y]=cirrdnPJ(x2,y2,rc2);
X2(t,:) = [ x y];
plot(x,y,'x');
end
X = [X1;X2];
pause;
hold on
[A,b,c,Aeq,beq]= lin(X',2);
lb = zeros(1,420);
ub = ones(1,420);
rtn = linprog(c,A,b,Aeq,beq,lb,ub);
zpq = rtn(1:400,:);
zpq
zz = reshape(zpq,[20,20]);
zz = zz';
yp = rtn(401:420,:);
ctr = find(yp);
ctr
% two centers
c1 = X(ctr(1),:);
c2 = X(ctr(2),:);
cc = [c1;c2];
plot(cc(:,1),cc(:,2),'*');
pause;
hold on
```

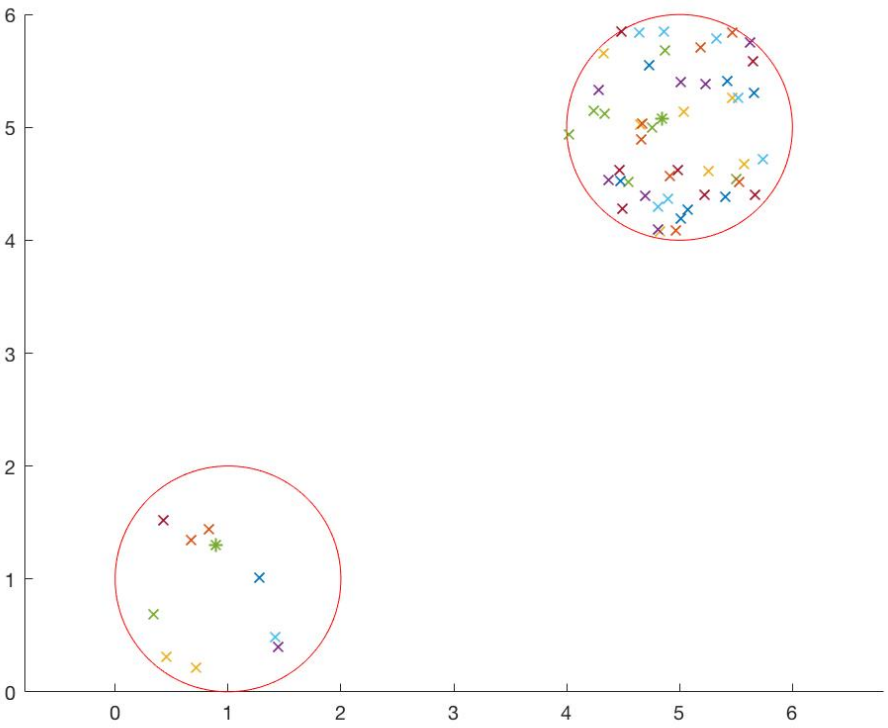
```
%Highlight the two centers
gp1 = zz(ctr(1),:);
g1 = find(gp1);
gg1 = X(g1,:);
plot(gg1(:,1),gg1(:,2),'o');
pause;
gp2 = zz(ctr(2),:);
g2 = find(gp2);
gg2 = X(g2,:);
plot(gg2(:,1),gg2(:,2),'o');
gp = [gp1;gp2];
```

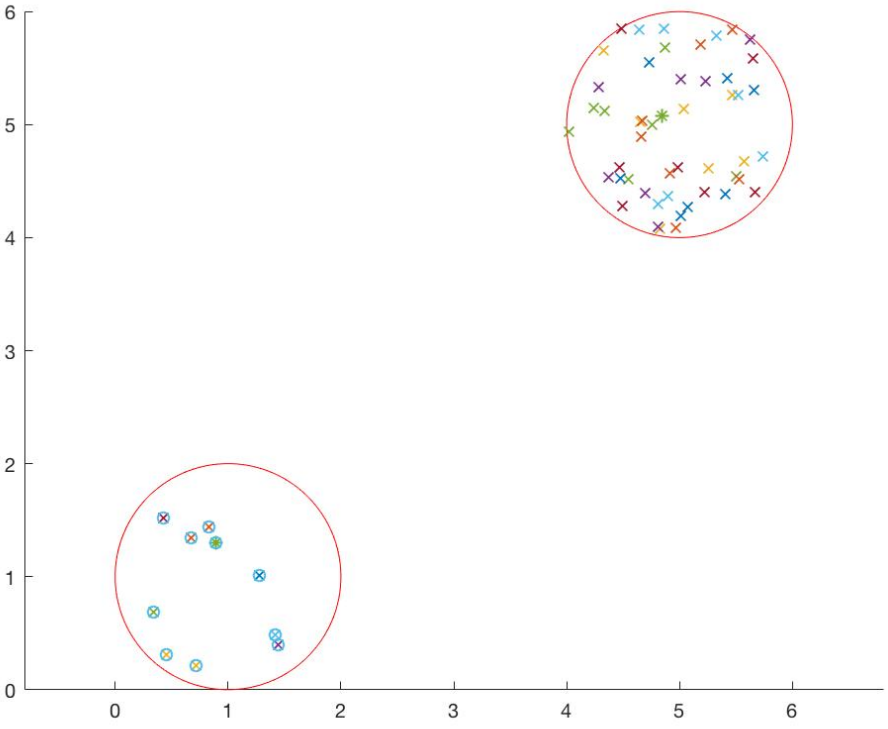
2.1.3 Two disjoint balls with same radius and different number of simulated points

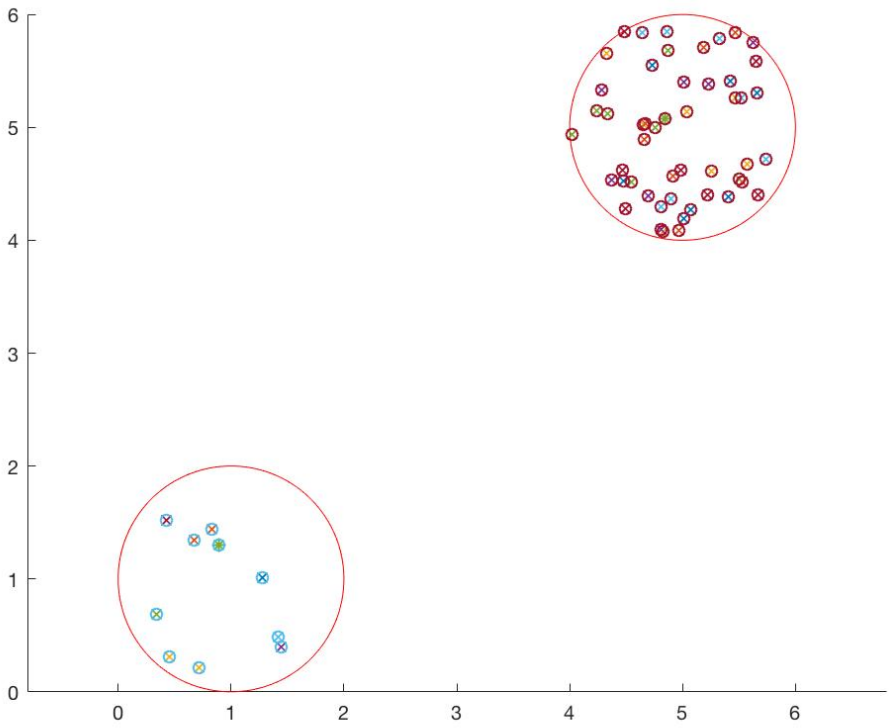
Num of Clusters k	Num of points in each cluster n
2	10 and 50

Centers	Radius	Integrality	Recover True Cluster?
(1,1) and (5,5)	1	Yes	Yes




0
1
2
3
4
5
6
0
1
2
3
4
5
6

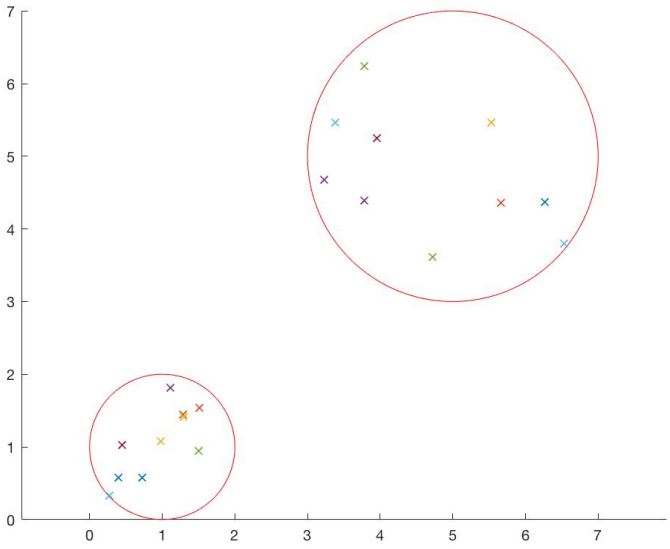

0
1
2
3
4
5
6
0
1
2
3
4
5
6

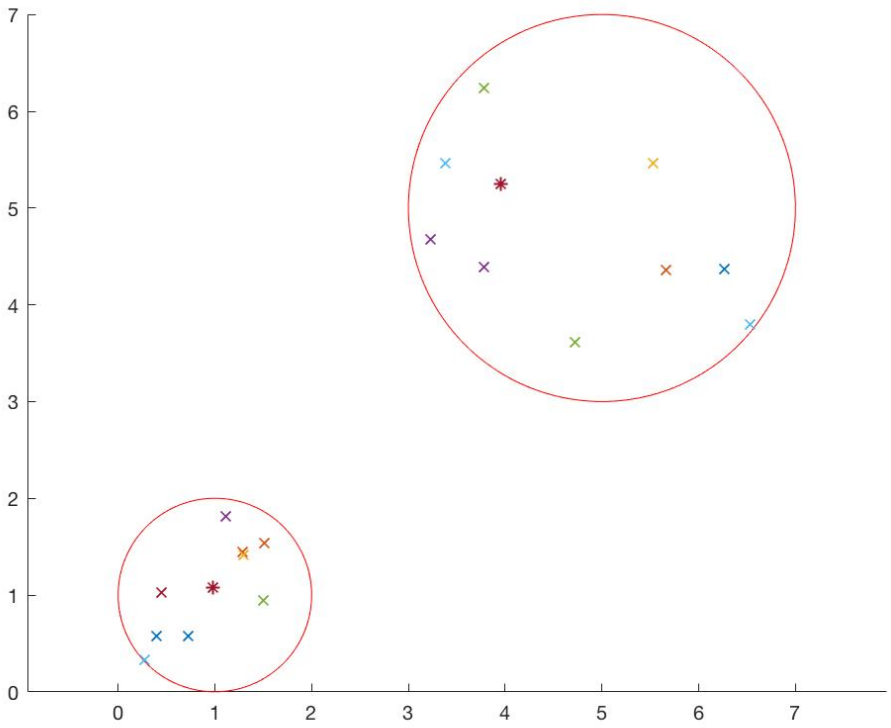


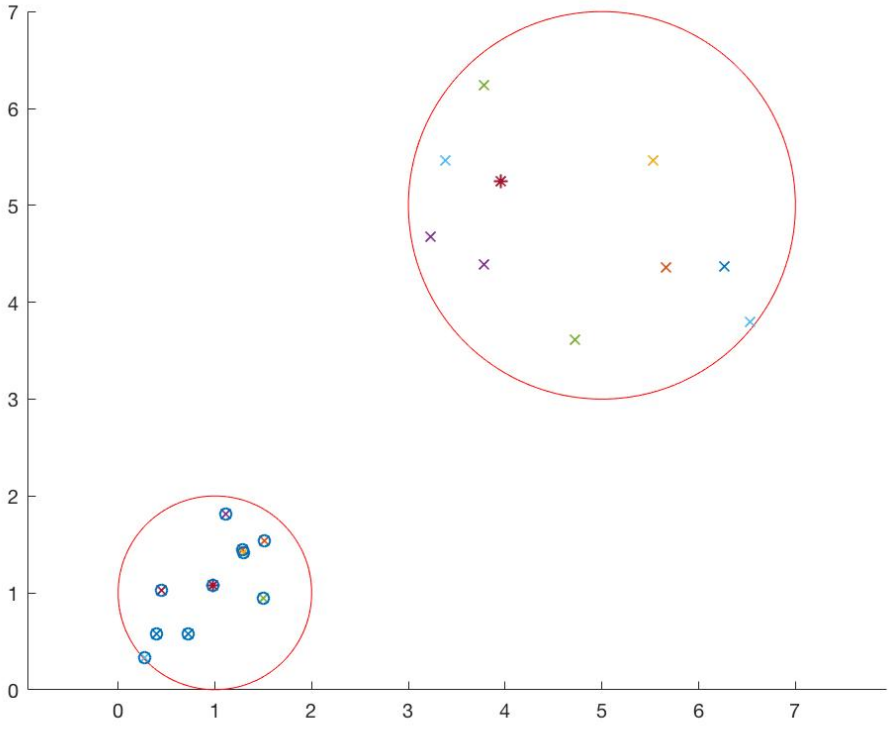
2.1.4 Two disjoint balls with different radius and same number of points

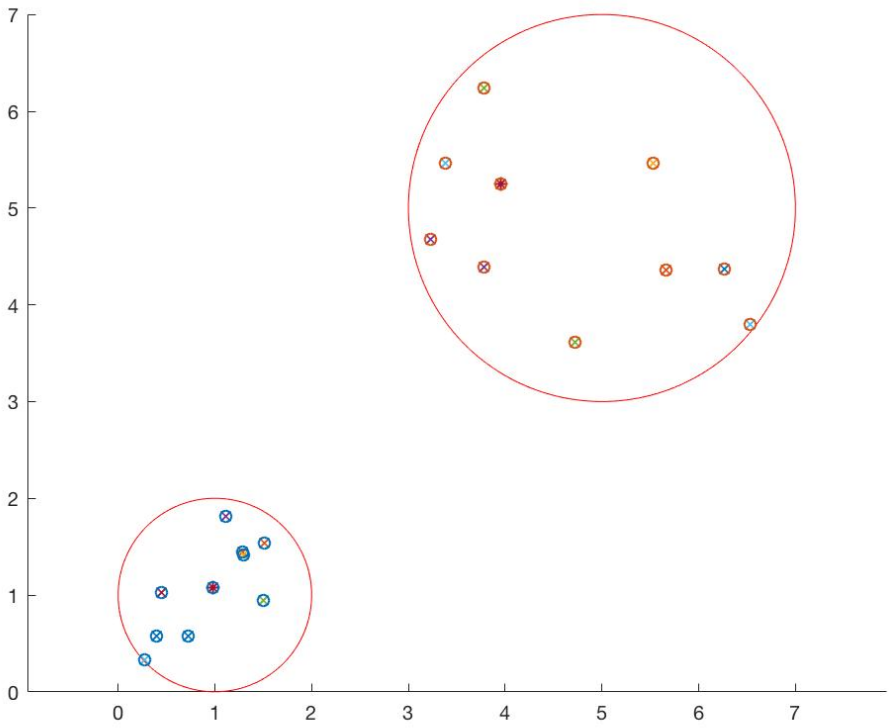
Num of Clusters k	Num of points in each cluster n
2	10 and 10

Centers	Radius	Integrality	Recover True Cluster?
(1,1) and (5,5)	1 and 2	Yes	Yes




0
1
2
3
4
5
6
7
0
1
2
3
4
5
6
7


0
1
2
3
4
5
6
7
0
1
2
3
4
5
6
7



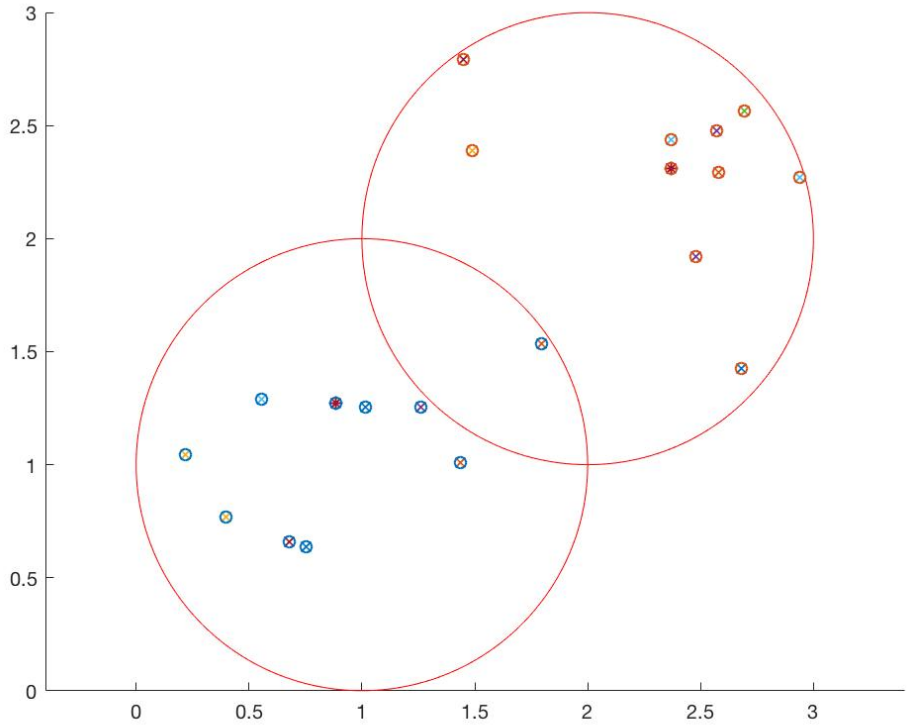
2.1.5 Two balls overlap with each other

Num of Clusters k	Num of points in each cluster n
2??	10 and 10

Centers	Radius	Integrality	Recover True Cluster?
(1,1) and (2,2)	1	Yes	Yes

For the first trial, there's only one point in the overlapped part. And the linprog calculation shows that it belongs to the first cluster, which is the circle centered

at $(1, 1)$, which is the true situation.



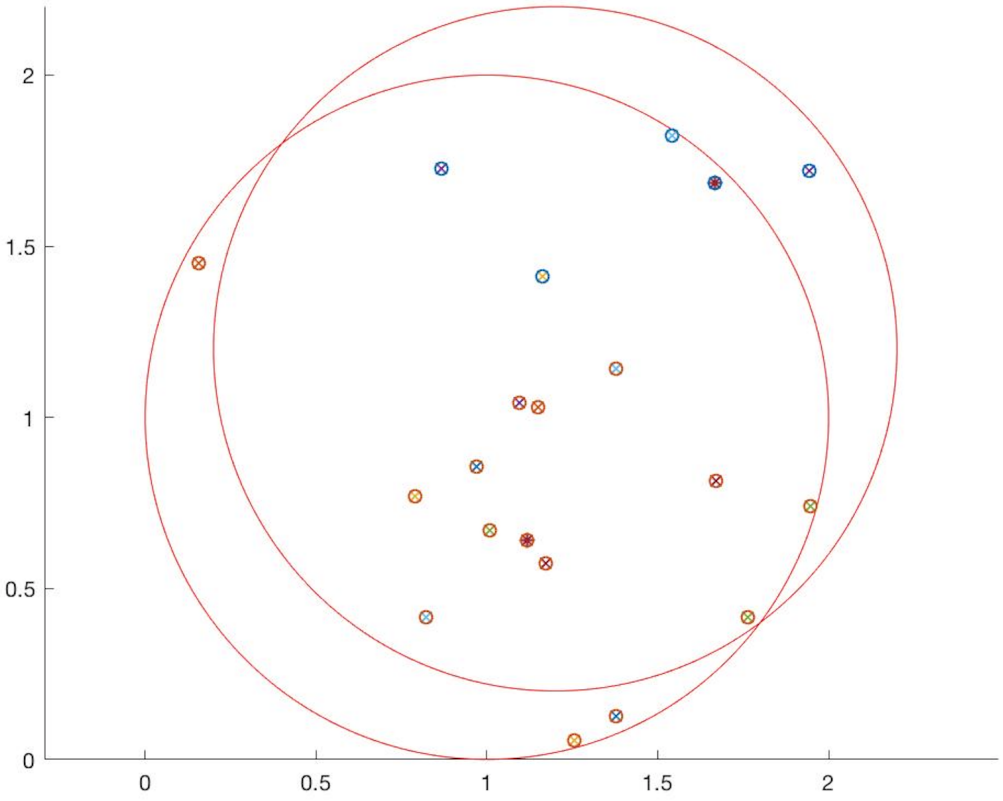
Trial 2 is a more interesting example:

Num of Clusters k	Num of points in each cluster n
2??	10 and 10

Centers	Radius	Integrality	Recover True Cluster?
(1,1) and (1.2,1.2)	1	Yes	??

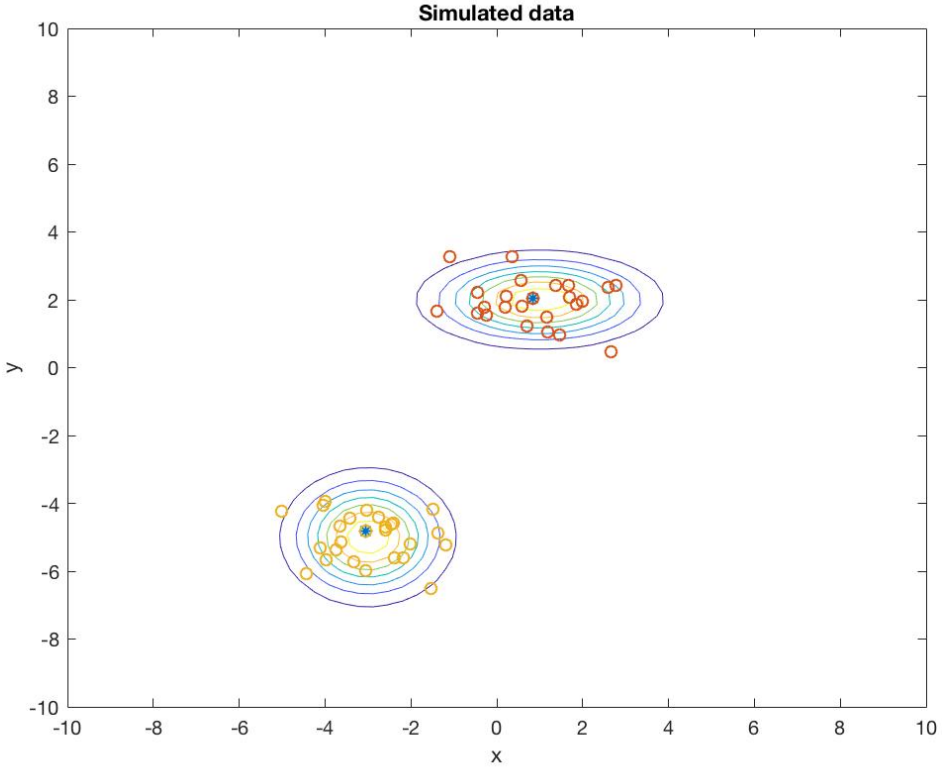
The result shows that the k-median clustering algorithm regards the points in the upper-right part as one cluster (*blue* points) and the points in the lower-left part as another (*red* points). And also the number of points in those two clusters are

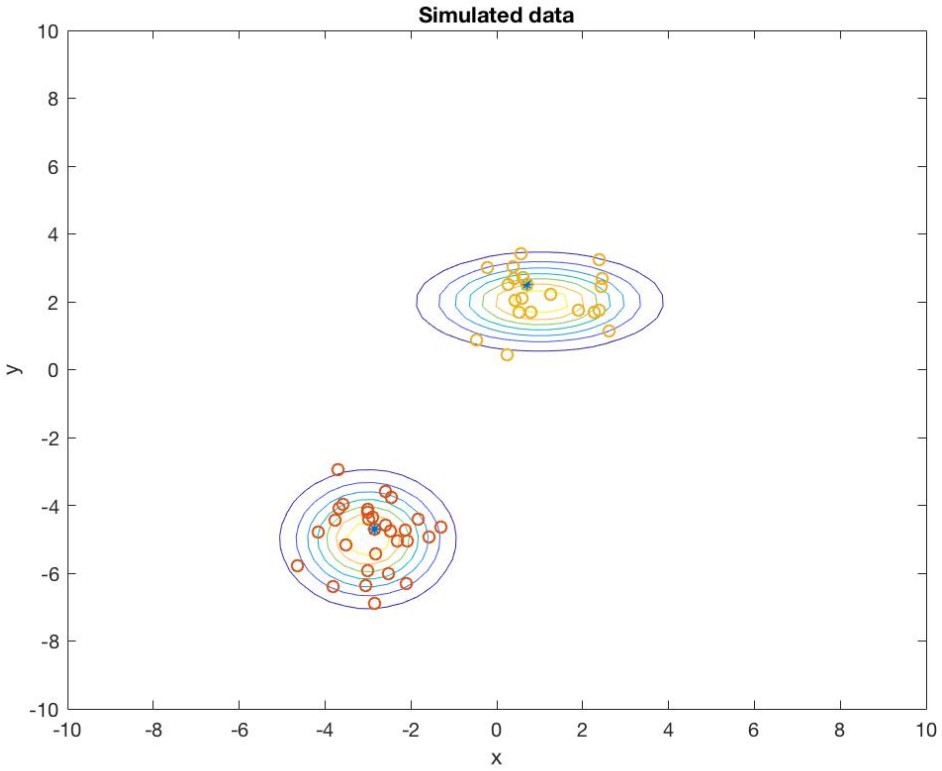
no longer even. (though the points generated from two circles are all 10.)

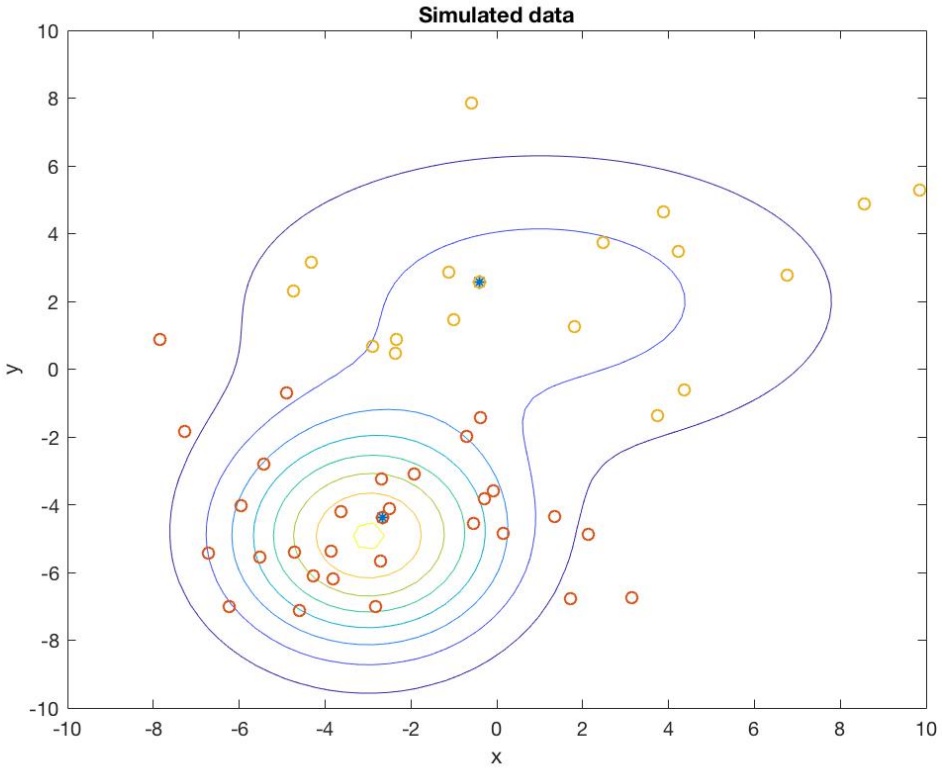


2.1.6 Gaussian Mixture Model

If our input data is generated from a GMM, what would be the possible results?







3 Week Three (July 9/July 11/July 13)

When the input data is still GMM, two mixed Gaussian distribution, but we increase the number of clusters from 2 to 3, then the solution will not always be integral.

For instance, look at the following trial. This is a GMM model, and the number of clusters is presented by k, and the number of points simulated in the GMM is N.

Trial	k	N	Num of centers	Integral solution ?	means	covariance
1	3	50	5	No	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
2	3	50	3	Yes	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
3	3	50	3	Yes	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
...						
...						
20	3	50	6	No	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
21	3	50	3	Yes	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
...						
23	3	50	6	No	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
...						
31	3	50	5	No	[1 2;-3 -5]	[25 0; 0 10] and [15 0; 0 5]
...						

From the computational result, under the condition that we draw 50 points from

the given GMM, the probability of getting a non-integral solution is nearly $4/30 = 0.1333$.

And one observation is intuitively, the GMM is two mixed Gaussian distribution, if we generate points from the GMM, then $k = 2$ makes more sense than the experimented $k = 3$. This is part of the reason, why $k = 2$, the solution we've got are all integral. (at least for the limited trials we have)

Then this brings about an interesting direction that given a random collection of points, how to decide what is the reasonable k ?

For instance, from a naive perspective, we may can imply that k needs to be 2 from the above trials, which makes the result more integral.

In fact,deciding the clustering factor k, is another minimization problem. In terms of the $k-$ median clustering problem, the k we are looking for is for making the sum of distances between points in each cluster and centers to be the minimum

$$\min \sum_{p,q \in P} d(p,q)$$

where P is the data set.

3.1 Experiments on testing the reasonable k for clustering

Input data: 100 uniformly distributed random points within a 5×5 square

Goal: Using the for loop on $k = 2, 10$, calculate the corresponding objective value to the $k - median$ LP and also the difference between each objective value. Observe the rapid decrease point, so that we can choose the reasonable k value for the clustering problem.

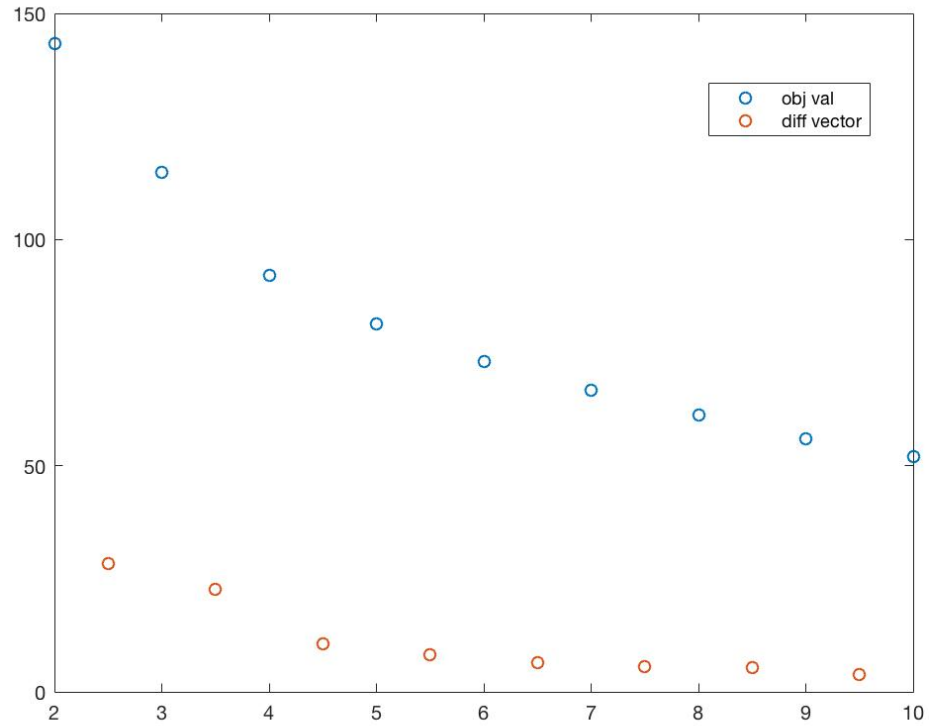
```
x=rand(1,100)*5;
y=rand(1,100)*5;
X = [x;y];
zoj = [];
scatter(x,y)
for k = 2:10
[A,b,c,Aeq,beq]= lin(X,k);
lb = zeros(1,10100);
ub = ones(1,10100);
[rtn,zval] = linprog(c,A,b,Aeq,beq,lb,ub);
zoj =[zoj zval];
end
zoj
% difference vector
zd = [];
n = length(zoj);
for i = 1:n-1
    diff = zoj(:,i) - zoj(:,i+1);
    zd = [zd diff];
end
zd
kv= [2 3 4 5 6 7 8 9 10];
plot(kv,zoj,'o');
```

```
hold on
dv = [2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5];
plot(dv,zd,'o');
```

plot and result:

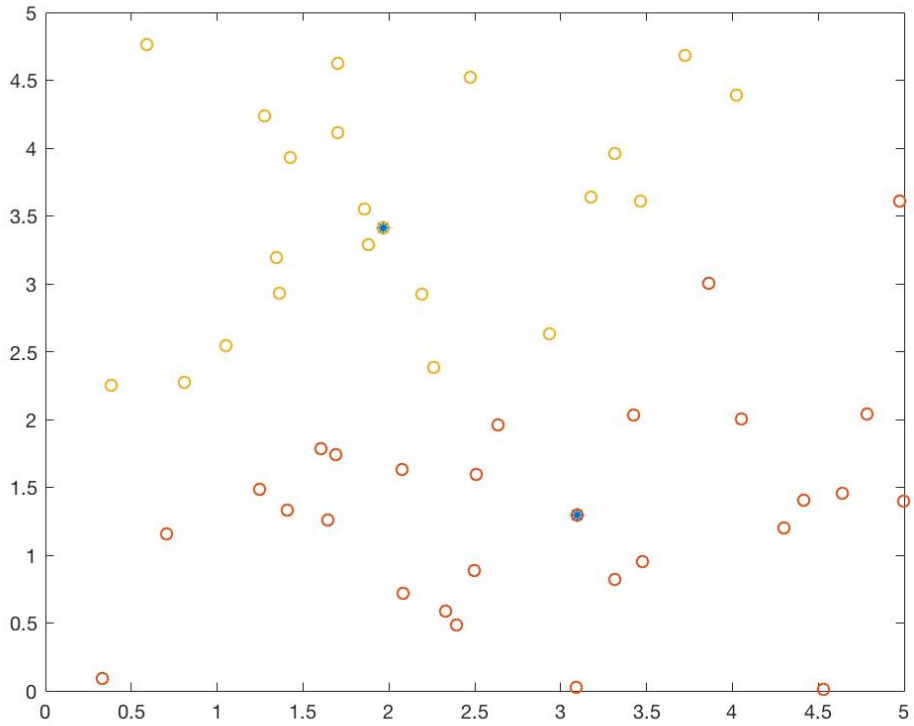
```
zoj =

  Columns 1 through 8

  143.2484  114.8455   92.1067   81.3842   73.0761   66.7145
61.1715   55.8725

  Column 9

   52.0723


zd =

   28.4030   22.7388   10.7225    8.3080    6.3617    5.5429
5.2990    3.8002
```

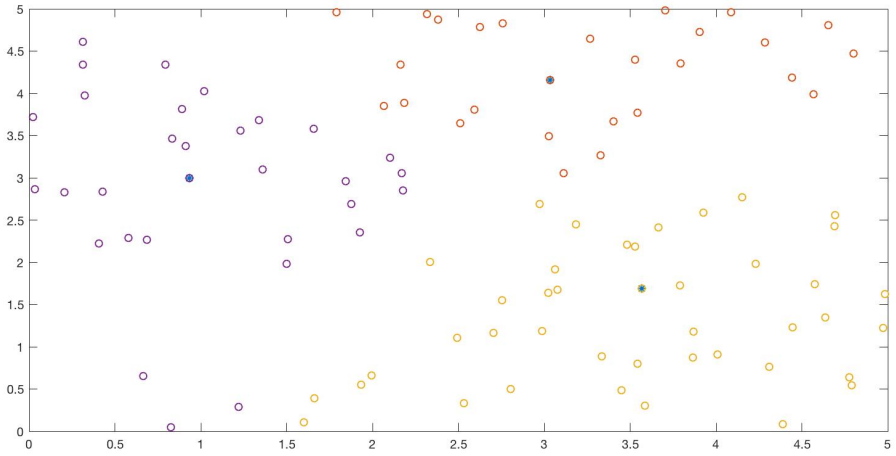
X-axis represents that $k = 2$ to 10, the y-axis represents the corresponding objective value.

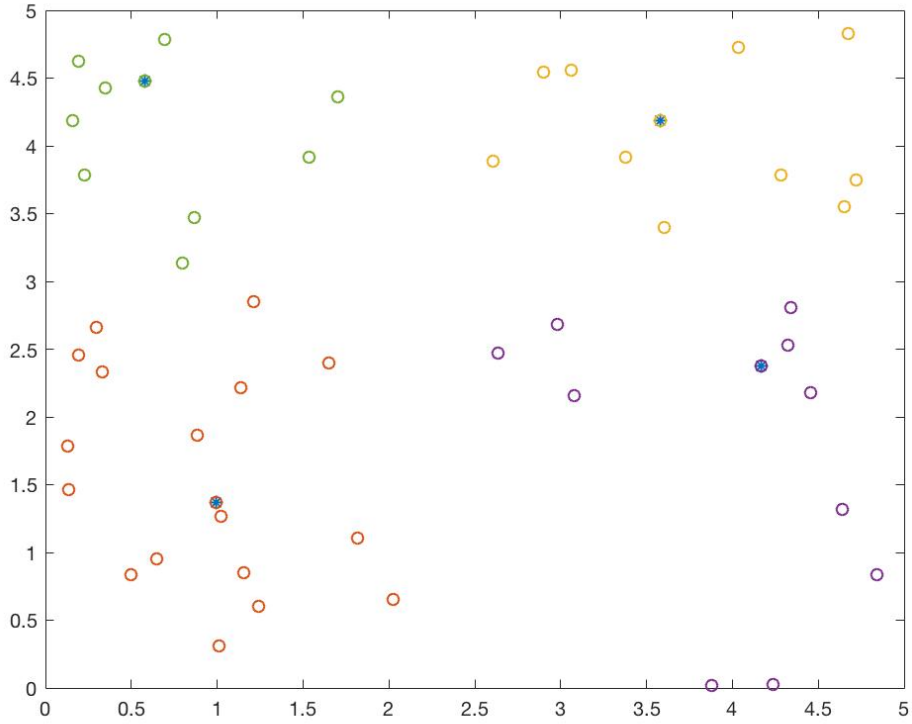


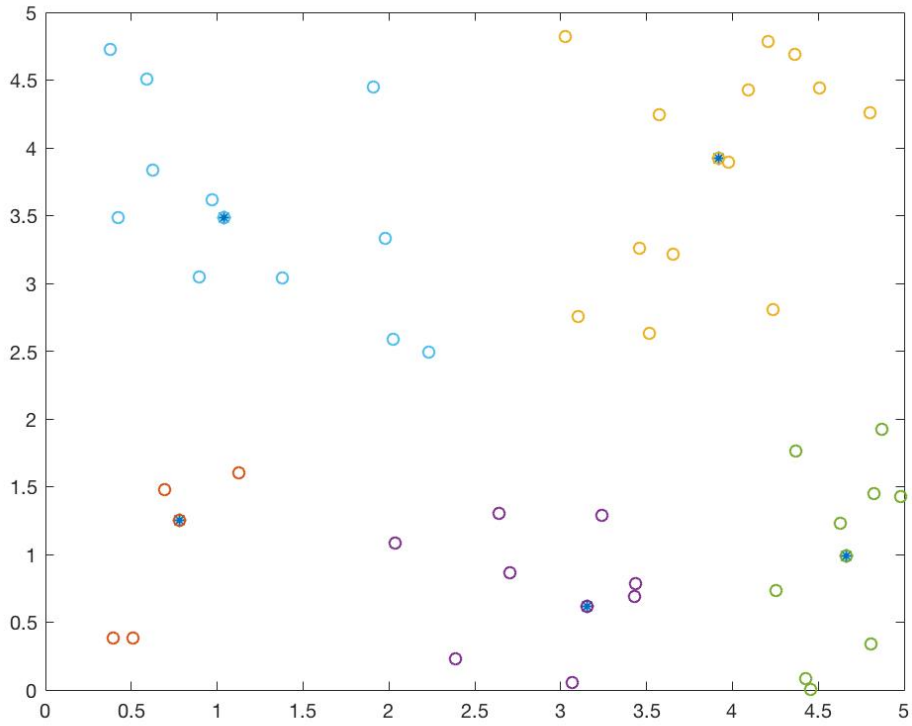
If we keep on doing the same experiment, simulate 10 times, the result all shows that for uniformly distributed points within a square, the sharpest drop is from $k = 2$ to $k = 3$.

```
zd =

   37.0052   20.0376   13.7121    9.3164    7.0416    4.5774
4.2166    3.4503

zd =

   29.3444   22.1111    9.6105    8.3565    6.0327    5.3245
4.0933    3.6177

zd =

   33.0127   19.8333    9.5861    8.4848    6.4476    5.7820
4.1499    3.9466
...
...
```




0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5


0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5



Based on the above experiments, we can probably get the potential conjecture as following:

Conjecture 1. Assume there's a series of data points, uniformly generated within a 5×5 square, the most reasonable clustering factor k for those points is 3.

3.2 Experiments gives integral solution or not?

First of all, I used the same input data with 100 uniformly chosen data points, with $k = 5$, the result is not always integral.

One of the non-integral solution is as the following, the y_p (which stands for the vector to indicate that whether the point p is the center of the cluster or not) is:

yp =

```
    0.2500
         0
         0
         0
         0
         0
         0
    0.2500
         0
         0
         0
```

0.2500
0
0.5000
0.5000
0
0
0
0
0
0
0
0
0.2500
0.7500
0
0
0.5000
0.5000
0
0
0
0
0
0
0
0
0
0
0
0.2500
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0

0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0
0.5000
0
0
0
0
0
0
0
0
0
0
0.5000
0
0
0
0
0
0
0
0
0
0

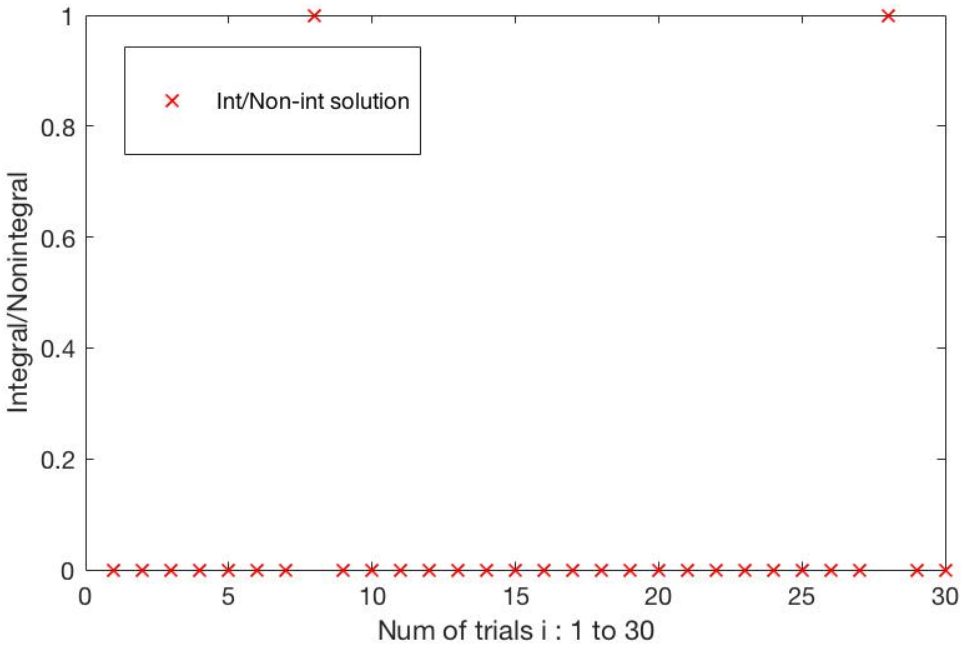
Non-integral entries appears are 1/4, 1/2 and 3/4.

Based this observation for $k = 5$ clustering , I started to do a new loop on $k = 3$, (according to the previous conjecture, supposed to be the best choice of k for uniformly simulated data points.)

I did the loop for 30 times, and use the indicate variable ig to represent whether the solution is integral or not. If it isn't, then $ig = 1$, if it is, then $ig = 0$.

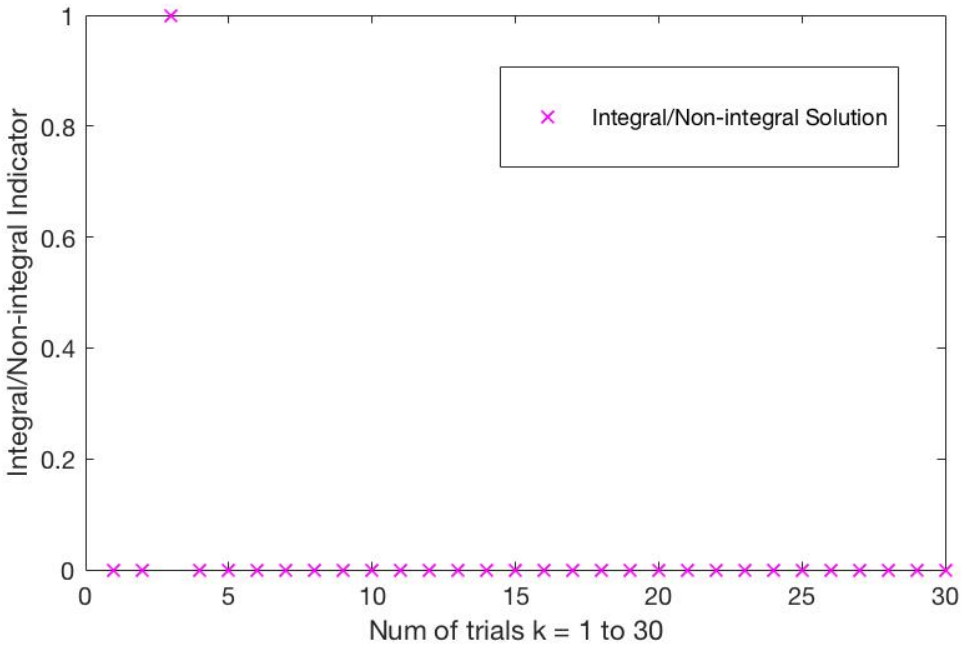
```
% k = 3 clustering
count = [];
ioni = [];
for i = 1:30
x=rand(1,100)*5;
y=rand(1,100)*5;
X = [x;y];
scatter(x,y)
[A,b,c,Aeq,beq]= lin(X,3);
lb = zeros(1,10100);
ub = ones(1,10100);
rtn = linprog(c,A,b,Aeq,beq,lb,ub);
zpq = rtn(1:10000,:);
zz = reshape(zpq,[100,100]);
zz = zz';
yp = rtn(10001:10100,:);
ctr = find(yp);
ctr
n = length(ctr);
% see whether the solution is integral or not. If the solution are all
% contained of integral entries, then n should be 3
if n == 3
    ig = 0;
else
    ig = 1; % non-integral solution
end
ig
ioni= [ioni ig];
count = [count i];
end
ioni
count
plot(count,ioni,'x');
```

The result shows that:



Observed result:**2 out of 30** trials are non-integral.

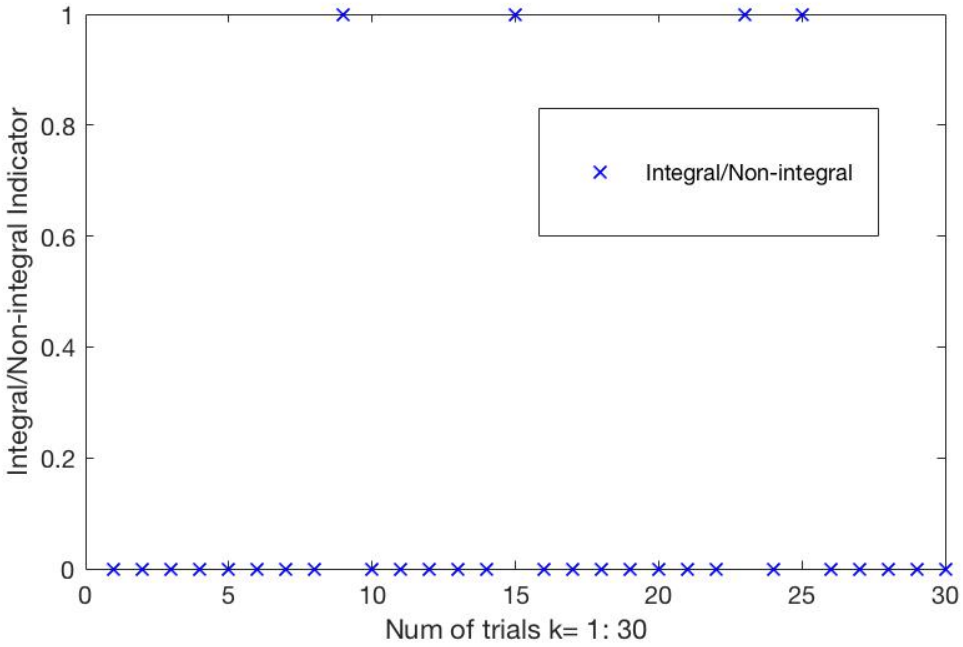
When $k = 4$ and $k = 5$, the experiment's result:



Only **1 out of 30** trials are non-integral.

For $k = 5$, slightly more non-integral solutions appear. (probably because, k

$= 5$ is a less reasonable choice compared to $k = 3, 4$)



Based on the above experiment results, we can have another potential potential conjecture:

Conjecture 2. Assume there's a series of data points, uniformly generated within a 5×5 sqaure, then there exists a high probability for the k-median LP relaxation to achieve integral solution, especially for k around the reasonable value.

4 Week Four (June 20)

Then as mentioned above, the non-integral solution may have a lot of forms, how to interpret the $0.5, 0.25, 0.75$ points?

Also how to prove the conjecture above?

4.1 Previous work

Recall from the previous work [ABC$^+$15], In Theorem 7,

Theorem. Let u be a probability measure in R^m supported in $B_1(0)$,continuous and rotationally symmetric with respect to 0 such that every neighborhood of 0 has positive measure. Then, given points $c_1, ..., c_k \in R^m$ such that $d(c_i, c_j) > 2$ if $i \neq j$, let μ_j be the translation of the measure μ to the center c_j.

Now consider the data set $A_1 = \{x_1\}^1$,..., $A_k = \{x_k\}^n$, each point drawn randomly and independently with probability given by $\mu_1, \mu_2, ..., \mu_k$ respectively.

Then , $\forall \gamma < 1$,$\exists N_0$ such that $\forall n > N_0$, the k-median LP is integral with probability at least γ.

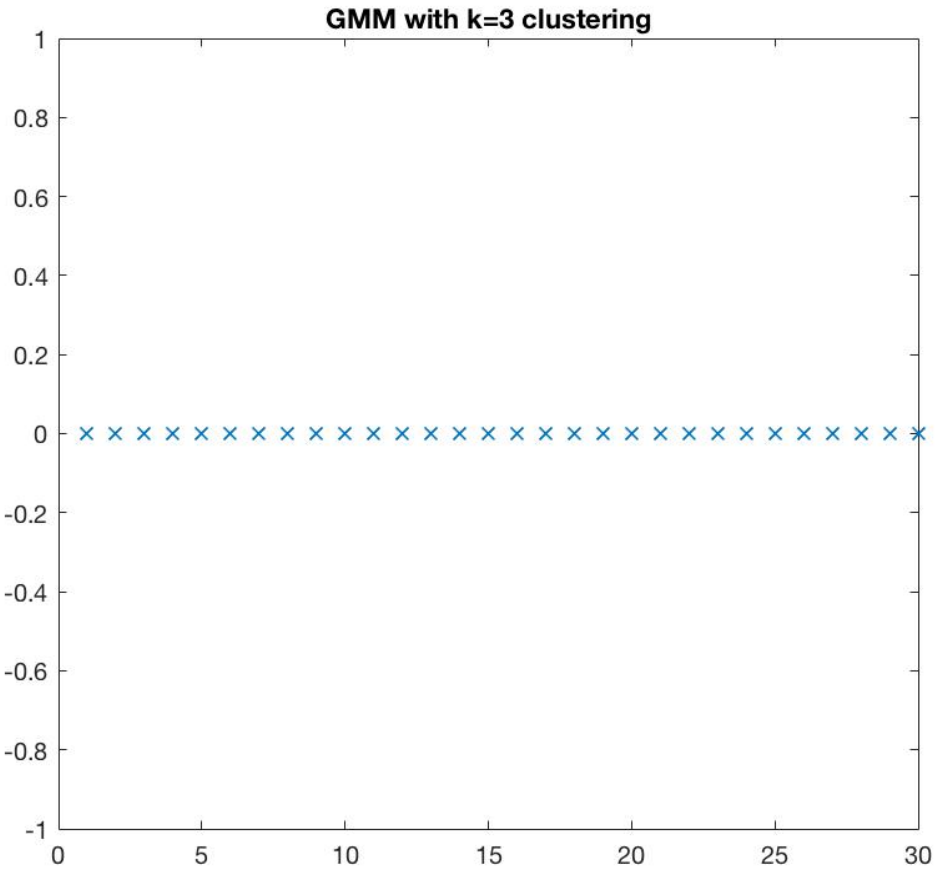
Also recall from the previous work [ABC+15], Lemma 5, when (5) satisfies, then the k-median LP is integral and the partition in clusters $A_1, ..., A_k$ is optimal. And more precisely, it requires the separation and center dominance conditions.

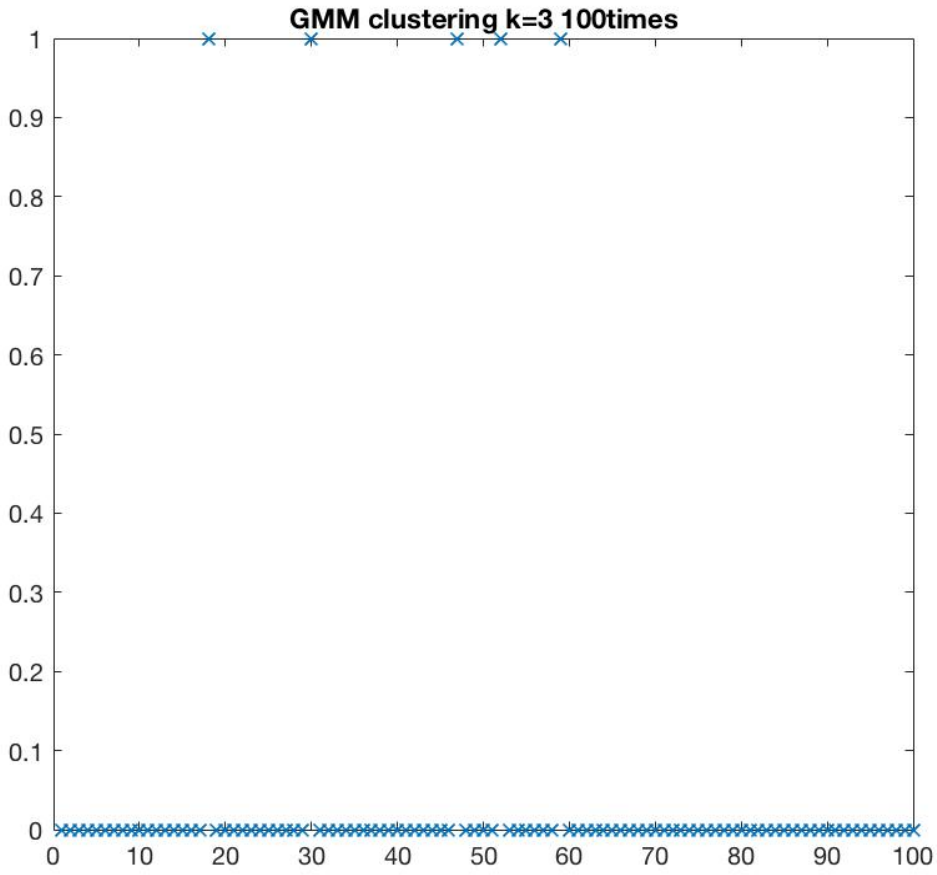
4.2 Input data points are arbitrarily drawn from uniform distribution

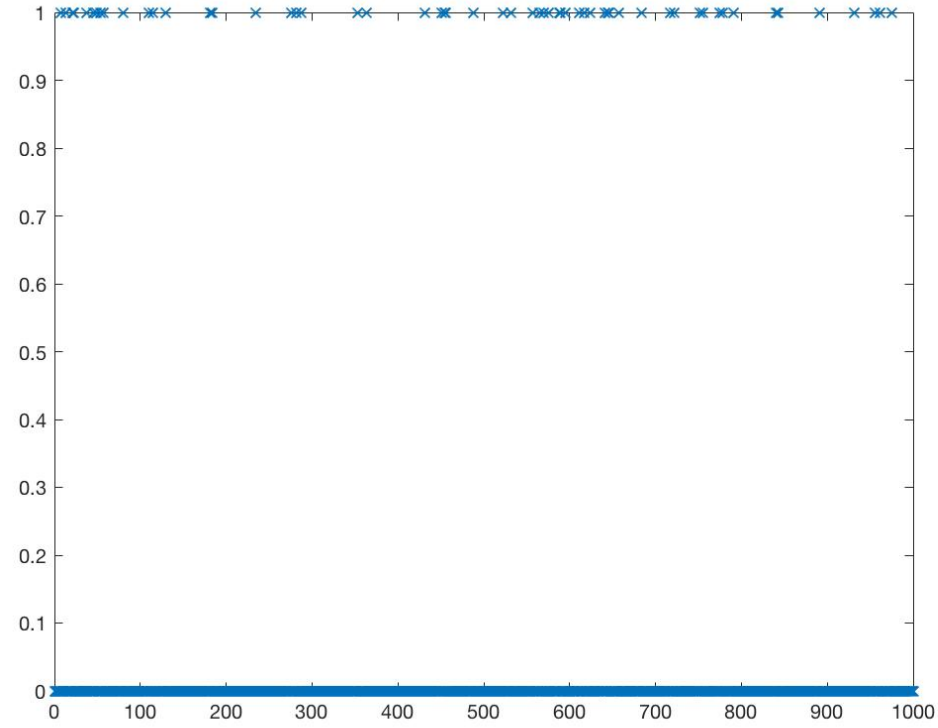
4.3 Input data points are drawn from Gaussian Mixture Model

One interesting observation is that, when use the same experiments as before, but apply on GMM, **the clustering calculation time** in Matlab is much faster.

```
% GMM_2, with k = 3 clustering
count = [];
ioni = [];
mu = [1 2;-3 -5]; %means
sigma = cat(3,[25 0;0 10],[15 0;0 5]); %covariance
p = ones(1,2)/2; %mixing proportions
gm = gmdistribution(mu,sigma,p);
for i = 1:30
X = random (gm,50);
[A,b,c,Aeq,beq]= lin(X',3);
lb = zeros(1,2550);
ub = ones(1,2550);
rtn = linprog(c,A,b,Aeq,beq,lb,ub);
zpq = rtn(1:2500,:);
zz = reshape(zpq,[50,50]);
zz = zz';
yp = rtn(2501:2550,:);
ctr = find(yp);
ctr
n = length(ctr);
% see whether the solution is integral or not. If the solution are all
% contained of integral entries, then n should be 3
if n == 3
    ig = 0;
else
    ig = 1; % non-integral solution
end
ig
ioni= [ioni ig];
count = [count i];
end
ioni
count
plot(count,ioni,'x');
```


GMM with k=3 clustering
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
5
10
15
20
25
30





When for the GMM, the number of experiment times increase to 1000, there are

57 non-integral solutions.

Conjecture 3. Assume there's a series of points simulated from a mixture of two Gaussian distributions, when the clustering factor $k = 3$, there's a high probability of getting integral solution from the k-median LP relaxation.

5 Week Five (June 23 / 25 /27)

5.1 Direction 1:Explore the dual variable α_q

5.1.1

Inspired by Lemma 5 in [ABC+15], the assumption made for the data points who has clustering structure, "Assume that we are given a clustered set $A_1, A_2, ...A_k$, with each $n_l, n_2, n_3, ..., n_k$ points, then if (5) condition is satisfied , then the k-median LP is integral and the partition is optimal."

Take the cases for the uniformly-simulated data points within a 5×5 square, If we only simulate 4 points in that square, and we let $k = 4$, then intuitively speaking, the evenly-divided clustering should be reasonable.

Then under this assumption, the condition of Lemma 5 would be satisfied.

Lemma. Consider sets $A_1, ..., A_k$ with $n_1, ..., n_k$ points respectively. If $\exists \alpha_1, ..., \alpha_k$ s.t. for each $s \in A_1 \cup ... \cup A_k$,

$$\frac{1}{k}([\sum_{i=1}^{k}[n_i\alpha_i - \min_{p \in A_i} \sum_{q \in A_i} d(p,q)]) \geq \sum_{q \in A_1}(\alpha_1 - d(s,q))_+ + ... + \sum_{q \in A_k}(\alpha_k - d(s,q))_+$$

then the k-median LP is integral and the partition in clusters $A_1, ..., A_k$ is optimal.

(Understanding of Lemma 5)

Then test for the case that four points stand for the four vertex of a 2×2 square within a 5×5 square. The distance is computed under the taxicab metric. (To make the distance to be integer numbers). Here $k = 2$

```
% The dual is doing maximizing problem
c = [ones(1,4) zeros(1, 16) -2]';
% there's no Aeq due to the reason that there's no equality condition
% x = [alpha_q, beta_pq,epsilon]';
O1 = ones(4,1);
I1 = eye(4);
A1 = kron(I1,O1);
I2 = eye(16);
A2 = [A1 -I2 zeros(16,1)];
O2 = ones(1,4);
A11 = kron(I1,O2);
L = [zeros(4,4) A11 -ones(4,1)];
A = [A2;L];
% The distance function are just calculated with the Euclidean metric
b1 = [0 2 2 4 2 0 4 2 2 4 0 2 4 2 2 0];
b1 = b1';
```

```
b = [b1; zeros(4,1)];
lb = [-10000*ones(1,4) zeros(1,16) -Inf]';
ub = 10000*ones(21,1);
Aeq = [];
beq = [];
[x, fval] = linprog(-c, A, b, Aeq, beq, lb, ub)
```

The result shows that $\alpha_1, \alpha_2, \alpha_3, \alpha_4 = 2$, and $\xi = 2$, which makes the dual objective value equal to 4.

Then compute the $c \times x_{feasible}$, we get the primal objective value to be 3. which makes sense, since the intended cluster solution is optimal if the corresponding LP objective value is less than or equal to the dual objective for some feasible point in the dual problem. (and since weak duality tells us that the primal value is always greater or equal than the dual value). Then by strong duality, we can get the **P = D**. Here, P is not equal to D.

When we make $k = 4$, the objective value becomes zero, and the feasible α gives us is also 2, which makes sense, since the α kind of represents the distances between the centers. When given four points, the four centers are respectively themselves, therefore, the sum of the distances between the centers of each cluster and the points are 0.

5.1.2

Inspired by the Theorem 7 in [ABC+15], as talked in the previous work in section 4. When the condition $d(c_i, c_j) > 2$,it's been proved that there exists some specific α value , which is $\alpha > 1$, then the expectation of the contribution function

$$P_i^{(\alpha,\ldots,\alpha)}(z)$$

attains its maximum at $z = c_j$.

Then doing our own experiment with,two overlapped uni-circles with distance between centers as$\sqrt{2}$:, and decreasing slowly.

5.2 Direction 2 : Change the metric into l_1

5.3 Direction 3: Weaken the condition that $d(c_i, c_j) > 2$

6 Week Six (June 30 / Aug 2/Aug 3)

6.1 Keep track of α and d between centers

1. one way is to set the α's equal within each cluster, and we will get $\alpha_1, \alpha_2, ..., \alpha_k$
2. the other way is to set all the αs equal. and then we will get $\alpha_1, \alpha_2, ...\alpha_k = \alpha$

$c_1 = (x_1, y_1)$	$c_2 = (x_2, y_2)$	n_1	n_2	$k = 2$	α	d
$(-\sqrt{2}/2, 0)$	$(\sqrt{2}/2, 0)$	50	50	/	0.8348	$\sqrt{2}$
$(-\sqrt{2}/4, 0)$	$(\sqrt{2}/4, 0)$	50	50	/	0.7197	$\sqrt{2}/2$
$(-\sqrt{2}/8, 0)$	$(\sqrt{2}/8, 0)$	50	50	/	0.6752	$\sqrt{2}/4$
$(-\sqrt{2}/16, 0)$	$(\sqrt{2}/16, 0)$	50	50	/	0.7598	$\sqrt{2}/8$
$(-\sqrt{2}/32, 0)$	$(\sqrt{2}/32, 0)$	50	50	/	0.7681	$\sqrt{2}/16$
$(-\sqrt{2}/64, 0)$	$(\sqrt{2}/64, 0)$	50	50	/	0.7579	$\sqrt{2}/32$
$(-\sqrt{2}/128, 0)$	$(\sqrt{2}/128, 0)$	50	50	/	0.7040	$\sqrt{2}/64$
$(-\sqrt{2}/256, 0)$	$(\sqrt{2}/256, 0)$	50	50	/	0.7131	$\sqrt{2}/128$
$(-\sqrt{2}/512, 0)$	$(\sqrt{2}/512, 0)$	50	50	/	0.6723	$\sqrt{2}/256$
$(-\sqrt{2}/1024, 0)$	$(\sqrt{2}/1024, 0)$	50	50	/	0.7175	$\sqrt{2}/512$
$(-\sqrt{2}/2048, 0)$	$(\sqrt{2}/2048, 0)$	50	50	/	0.7055	$\sqrt{2}/1024$
$(-\sqrt{2}/4096, 0)$	$(\sqrt{2}/4096, 0)$	50	50	/	0.8117	$\sqrt{2}/2048$
$(-\sqrt{2}/8192, 0)$	$(\sqrt{2}/8192, 0)$	50	50	/	0.7233	$\sqrt{2}/4096$
$(-\sqrt{2}/16384, 0)$	$(\sqrt{2}/16384, 0)$	50	50	/	0.8334	$\sqrt{2}/8192$

And all of the above trials can achieve *integral* solution.

This is just a single trial for different d, the value of α will vary if we try several times. However, the range for the α value is within $(0.55, 0.95)$.

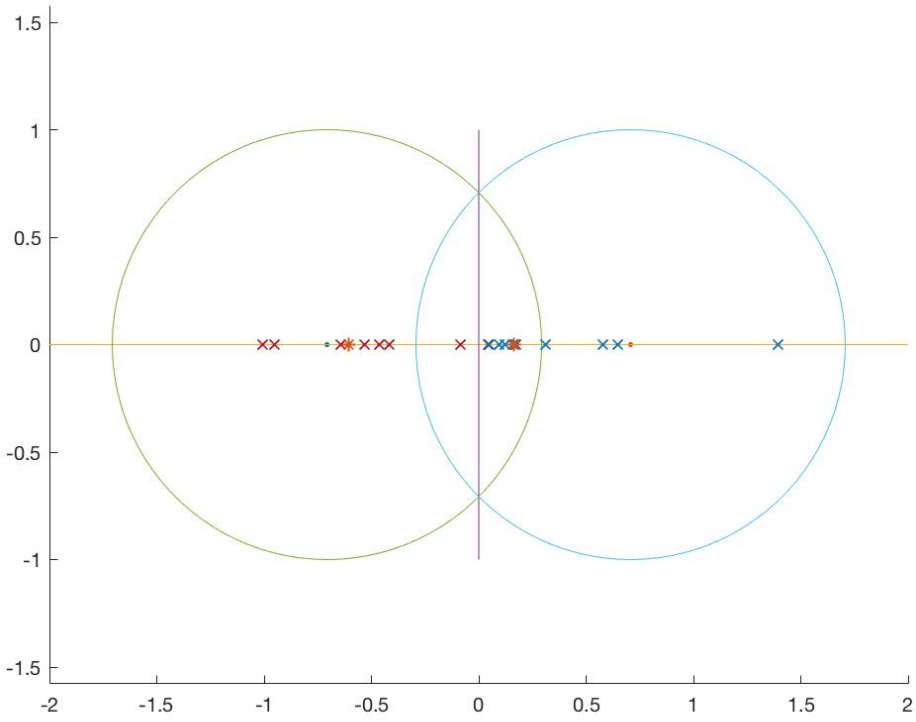
6.2 Dual certificate α

6.2.1 observations on two overlapped circles

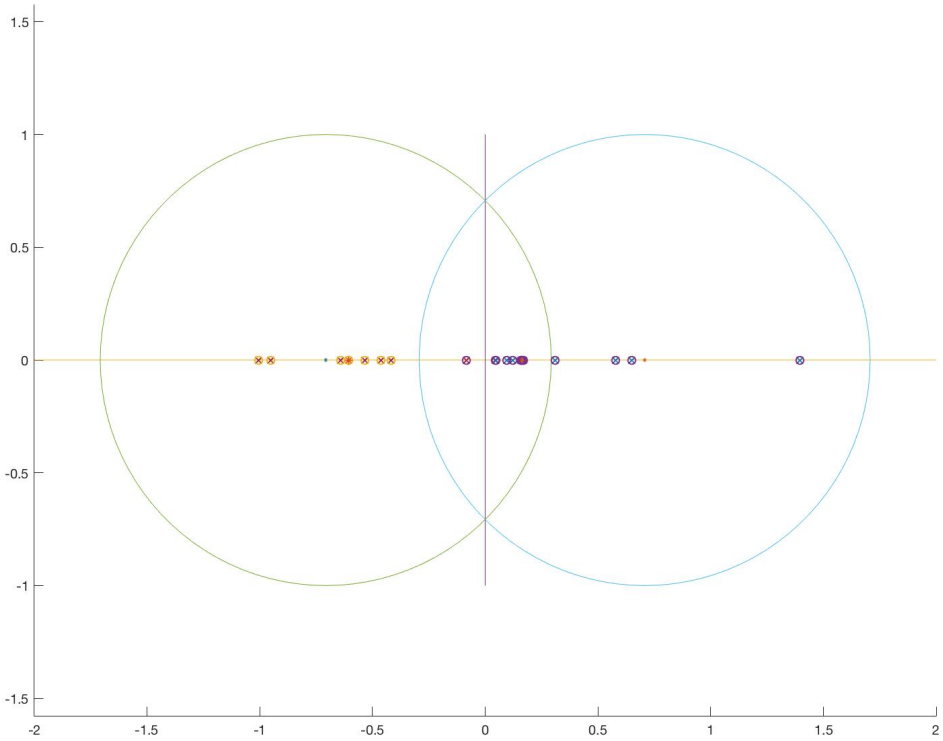
Case I: when the distance between the centers is large ($d = \sqrt{2}$)

First of all, we tried the simple example, which we simulated uniformly data points on the line $y = 0$. And by intuition, the natural way to do clustering is to cluster by the *perpendicular bisector line*. And we define it as the true cluster("ground truth") of the overlapped circle situation.

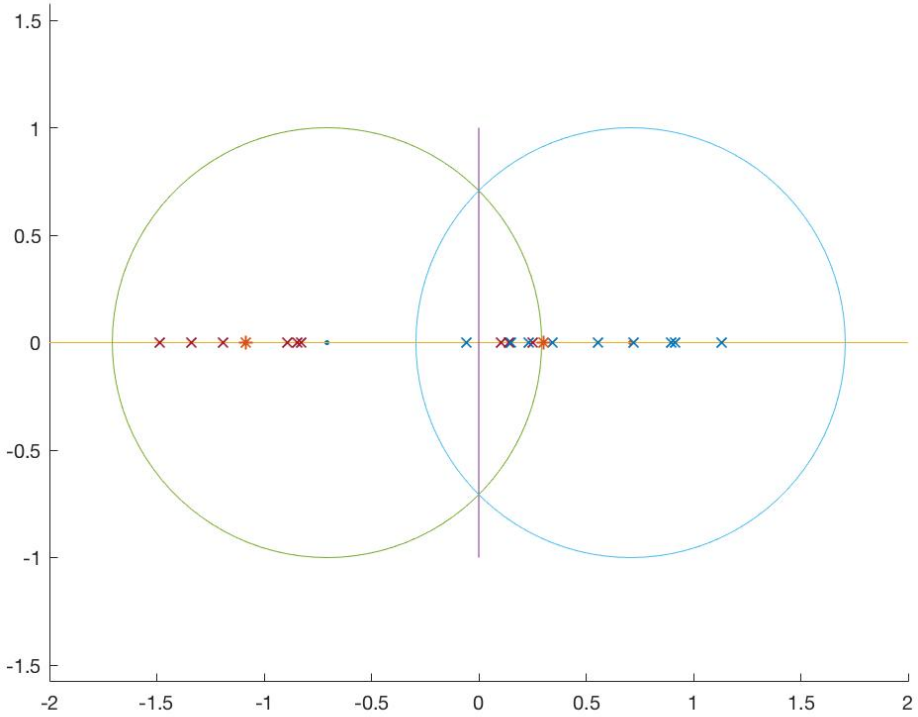
observation results:



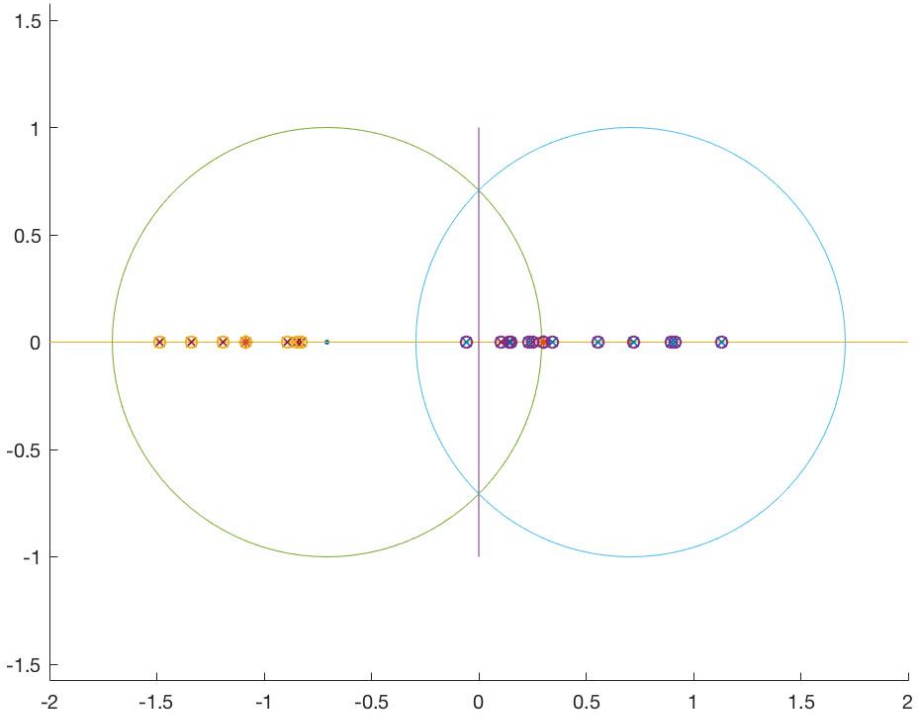
This shows the simulated points and also the clustered centers $'*'$.



Yellow small circles represent the clustered A_1 and the purple small circles represent the clustered A_2. $\alpha = 0.2429$ for this example.

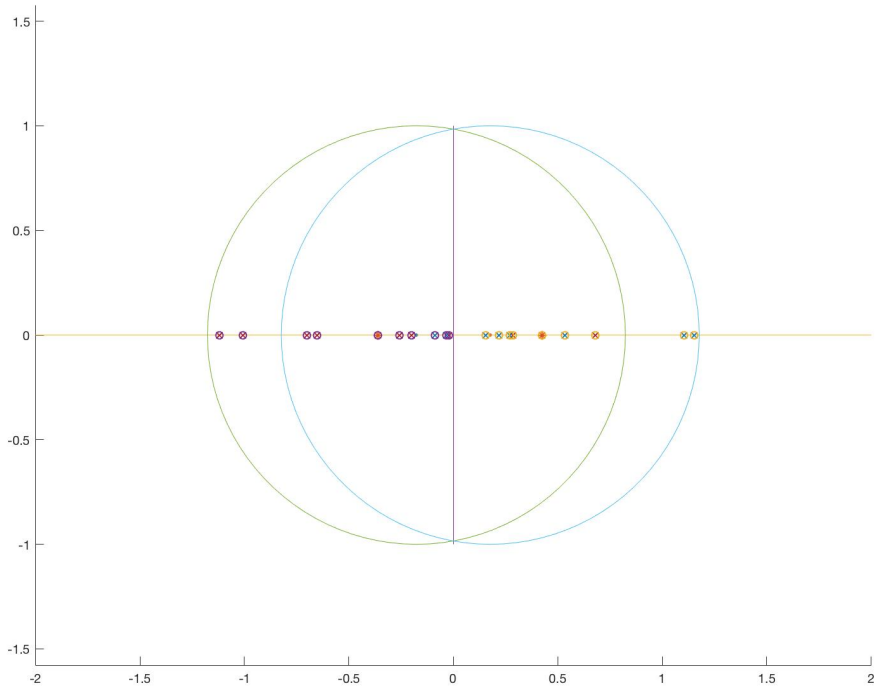


Before the clustering,and the two centers are picked.



And the α value for this example is 0.4913.

Case II: When the two circles overlap with each other a lot $d = \sqrt{2}/4$



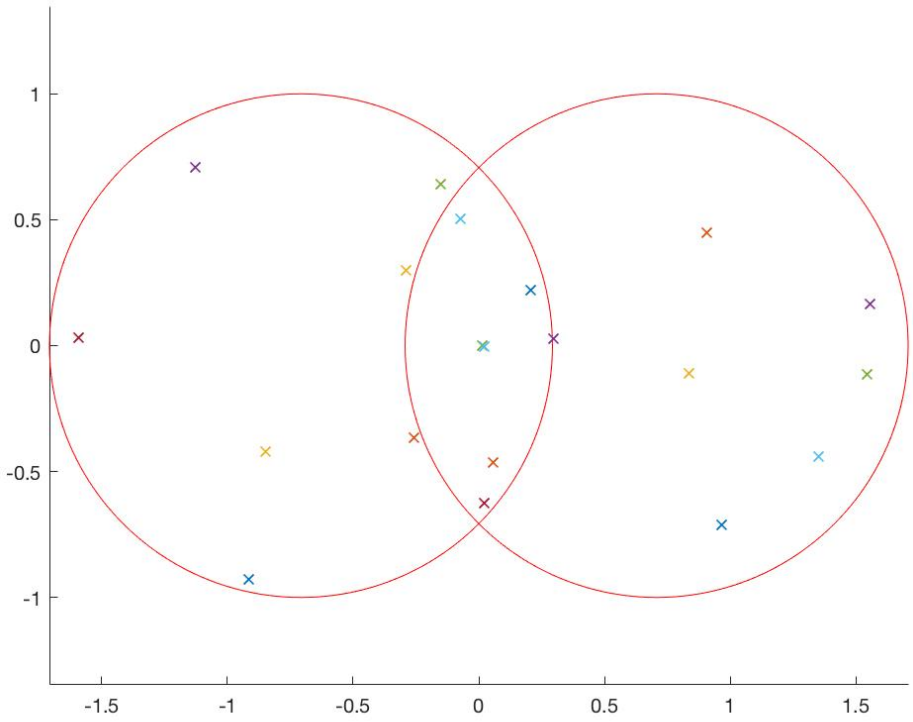
In this case, the two clusters are roughly evenly distributed on two sides of the perpendicular bisector line. The α value is 0.3587.

```
clf
axis equal
hold on
c_ix = - sqrt(2)/8;
c_iy = 0;
c_jx = sqrt(2)/8
;
c_jy = 0;
plot(c_ix,c_iy,'.');
hold on
plot(c_jx,c_jy,'.');
hold on
X = -2:0.1:2;
Y = 0 * ones(size(X));
plot(X, Y);
hold on
Y1 = -1:0.1:1;
X1 = 0 * ones(size(Y1));
plot(X1, Y1);
```

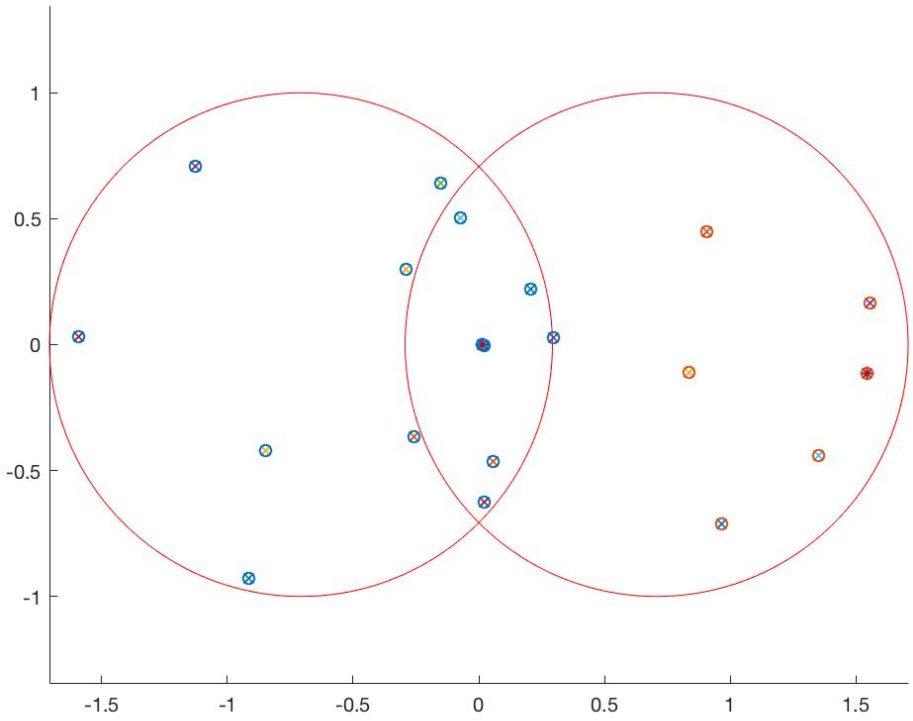
```
hold on

radius = 1;
th = 0:pi/50:2*pi;
xunit = radius * cos(th) + c_ix;
yunit = radius * sin(th) + c_iy;
h = plot(xunit, yunit);
hold on
th = 0:pi/50:2*pi;
xunit1 = radius * cos(th) + c_jx;
yunit1 = radius * sin(th) + c_jy;
h = plot(xunit1, yunit1);
hold on

a1 = c_ix-1;
b1 = c_ix+1;
r1 = (b1-a1)*rand(10,1) + a1;
r1_range = [min(r1) max(r1)];
r1 = [r1 zeros(10,1)];
plot(r1(:,1),r1(:,2),'x');
hold on

a2 = c_jx-1;
b2= c_jx+1;
r2 = (b2-a2)*rand(10,1) + a2;
r2_range = [min(r2) max(r2)];
r2 = [r2 zeros(10,1)];
plot(r2(:,1),r2(:,2),'x');
pause;
% generate 20 points in total
r = [r1;r2];

[A,b,c,Aeq,beq]= lin(r',2);
lb = zeros(1,420);
ub = ones(1,420);
rtn = linprog(c,A,b,Aeq,beq,lb,ub);
rtn
zpq = rtn(1:400,:);
zpq
zz = reshape(zpq,20,20);
zz = zz';
yp = rtn(401:420,:);
ctr = find(yp);
ctr
% two centers
c1 = r(ctr(1),:);
c2 = r(ctr(2),:);
cc = [c1;c2];
```

```
plot(cc(:,1),cc(:,2),'*');
pause;
hold on
%Highlight the two centers
gp1 = zz(ctr(1),:);
g1 = find(gp1);
gg1 = r(g1,:);
plot(gg1(:,1),gg1(:,2),'o');
pause;
hold on
gp2 = zz(ctr(2),:);
g2 = find(gp2);
gg2 = r(g2,:);
plot(gg2(:,1),gg2(:,2),'o');
gp = [gp1;gp2];

%test the dual certificate
[A1,b1,c1,Aeq1,beq1,lb1,ub1]= lindual(r',2);
[rtn1,fval] = linprog(-c1,A1,b1,Aeq1,beq1,lb1,ub1);
rtn1(1:20)
fval
```

Case III: $d = \sqrt{2}$ **points are simulated within the whole circles**



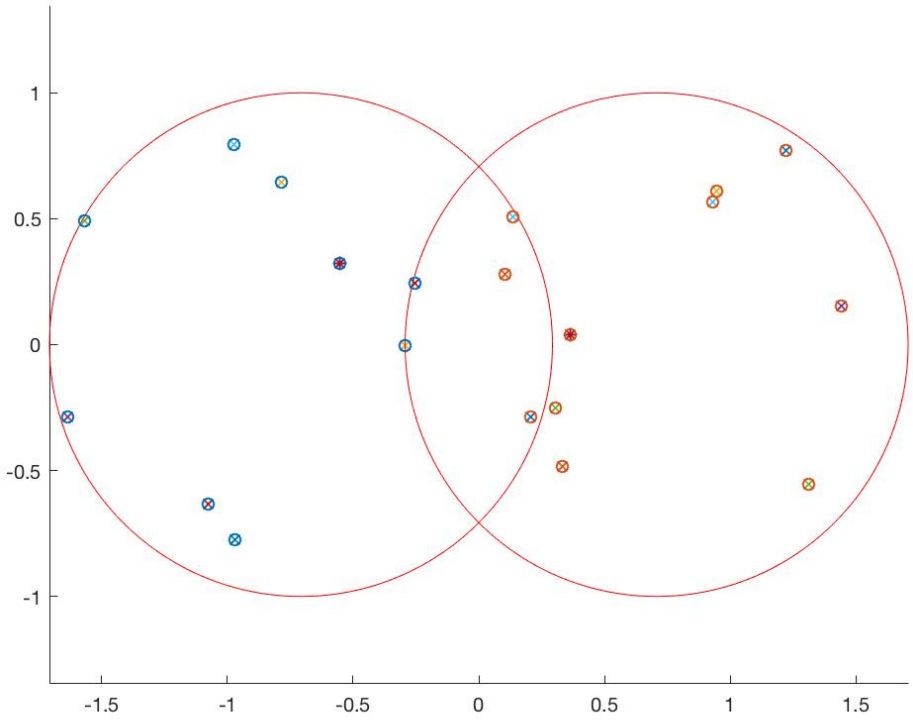
after clustering



The α value for this example is 0.6657.

Second trial,the $\alpha = 0.8722$,increased,with distance between the points and

the clustered centers getting bigger.



Based on the observation above, and the insights given in the section 7 in [ABC+15], the difficulty when studying the overlapped two balls, is because there's no 'ground truth' that whether we recover the true cluster or not.

However, based on our experiment, the α still seems to be some kind of distance thresholds.

7 Week Seven (Aug 7 / Aug 9)

7.1 Two overlapped intervals

Since it's really difficult to explore the dual certificate in the overlapped two circles,i.e. in the R^2 situation, then it would be an interesting question to just investigate the R^1 situation, i.e. the two overlapped intervals.

Question : Is the solution for $k-$ median LP relaxation for the two overlapped intervals always integral or we get the integral solution with high probability?

7.2 Experiment results

First trial, we used uniform distribution, draw 10 points respectively from the two intervals:

$$[-\sqrt{2}/8 - 1, -\sqrt{2}/8 + 1], \textit{and } [\sqrt{2}/8 - 1, \sqrt{2}/8 + 1]$$

Surprisingly, after a few trials the solution is always integral.

Then modify the code a little bit, we can do experiments for 30 times to see for $d = \sqrt{2}/4$ and overlapped a lot situation, the solutions' behavior.

```
% run the 1D experiments for several times to see the solutions are alwa
% integral or integral with high probability?
count = [];
ioni = [];
for i = 1:30
c1 = - sqrt(2)/8;
a1 = c1-1;
b1 = c1+1;
r1 = (b1-a1)*rand(10,1) + a1;
r1_range = [min(r1) max(r1)];
c2 = sqrt(2)/8;
a2 = c2-1;
b2 = c2+1;
r2 = (b2-a2)*rand(10,1)+ a2; % rand function is simulating pts uniformly
r2_range = [min(r2) max(r2)];
%%test for integral solutions? (with the 20 generated points)
r = [r1;r2];
[A,b,c,Aeq,beq]= linforoneD(r',2);
lb = zeros(1,420);
ub = ones(1,420);
rtn = linprog(c,A,b,Aeq,beq,lb,ub);
zpq = rtn(1:400,:);
zz = reshape(zpq,20,20);
zz = zz';
yp = rtn(401:420,:);
ctr = find(yp);
ctr %integral solution?
% two centers
n = length(ctr);
if n == 2
    ig = 0;
else
    ig = 1; % non-integral solution
end
ioni= [ioni ig];
count = [count i];
end
plot(count,ioni,'x');
```

The results are 30/30 are integral solutions.

If we increase the number of experiments,(I tried 100,1000) the result is 100/100 and 1000/1000 are integral solutions.

So maybe a possible conjecture is for the $\mathbb{R}^1$ case, $k-$median LP clustering on overlapped intervals always gives integral solution.

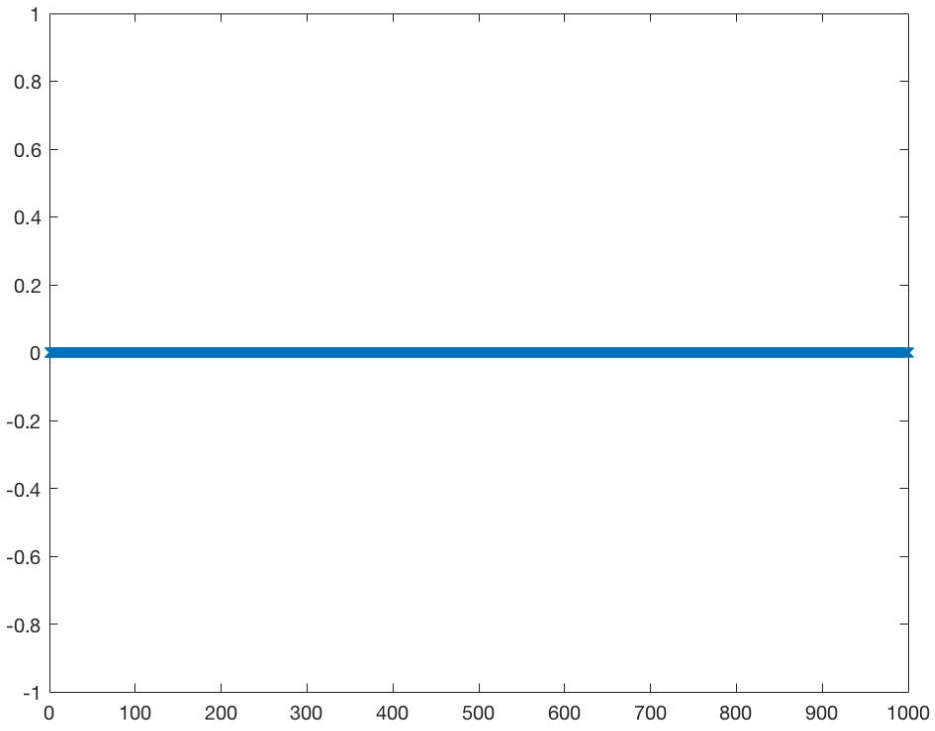
With the distance between two centers $d = \sqrt{2}/4$ unchanged, and if we decrease/increase the overlapped part: i.e. $(c_1, c_2 = \pm\sqrt{2}/8)$

$$[c_1 \pm 0.2, c_2 \pm 0.2] always\ integral$$

$$[c_1 \pm 1.1, c_2 \pm 1.1] always\ integral$$

$$[c_1 \pm 1.5, c_2 \pm 1.5] always\ integral$$

$$[c_1 \pm 2.0, c_2 \pm 2.0] always\ integral$$

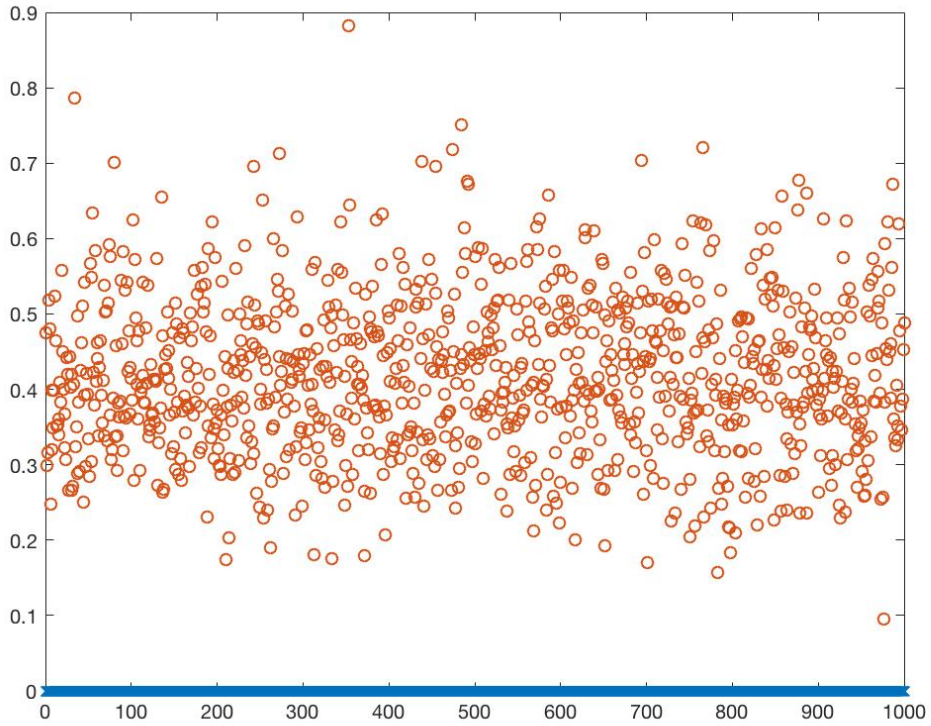


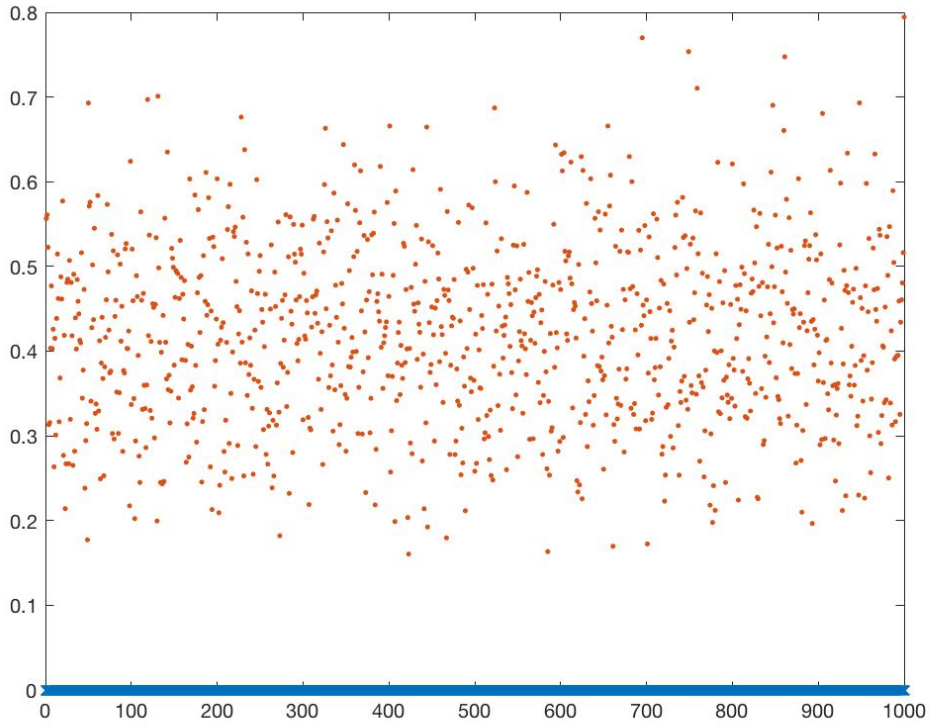
If we adjust the d between two centers, the results are still always integral.

7.3 If the centers are not symmetric to the origin?

7.4 α Dual variable observations on $\mathbb{R}^{\mathbb{1}\!\!\mathbb{1}}$

For the experiment, the dual variables α value: For 1000 number of experiments, all of them are integral.





8 Week Eight (Aug 14/Aug 16)

8.1 1-D Proof of Integrality based on Complementary Solutions/ Dual Variables for Primal LP

First consider the following concept: *complementary slackness*

$$(P) : \min c^T x$$

$$s.t. Ax = b, x \geq 0,$$

$A \in R^{m \times n}$, $b \in R^m$, $c \in R^n$ $m \leq n$. Solution vector for (P) : $x \in \mathbb{R}^n$

$$(D) : \max b^T y$$

$$s.t. A^T y \leq c$$

or

$$\max b^T y$$

$$s.t. A^T y + s = c$$

$$s \geq 0$$

$s \in \mathbb{R}^n$ $y \in \mathbb{R}^m$. Solution vector for (D): $(s, y) \in \mathbb{R}^{n+m}$. A pair of feasible solutions x,(s, y) satisfies **complementary slackness** provided that

$$x_1 s_1 = 0$$

...

$$x_n s_n = 0$$

that is for each i, $x_i = 0$ or $s_i = 0$. so: if $x_i > 0$, then $s_i = 0$;
if $s_i > 0$, then $x_i = 0$.

Then back to our proof of integrality.

If we define our dual variables α_q to be the difference for each point q to its nearest assigned center.

For instance, if we are in a case of $k = 2$ and in R^1, and the two centers we got after clustering algorithm by solving linprog, is c_i and c_j. Here, the optimal solution after k-median clustering algorithm, we assume that : pick the median of the data points,and cut it into half from that median point. And then if the q we choose is near c_i, then $\alpha_q = d(c_i, q)$.

Then if we look up the dual constraints of our k-median linear program,

$$\max_{\alpha \in \mathbb{R}^n} \sum_{q \in P} \alpha_q - k\xi$$

$$s.t. \alpha_q \leq \beta_{pq} + d(p, q), \forall p, q \in P$$

$$\sum_q \beta_{pq} \leq \xi, \forall p \in P$$

$$\beta_{pq} \geq 0, \forall p, q \in P$$

For the first constraint, $\alpha_q \leq \beta_{pq} + d(p, q), \forall p, q \in P$,

if $a_q - d(p, q) < 0$, strictly smaller than 0, then $\beta_{pq} = 0$, then variable β must equal to zero.

if $\alpha_q - d(p, q) > 0$, then $\beta_{pq} \geq \alpha_q - d(p, q)$

Therefore, if we sum up for all β_{pq} $\forall p$, (and also for each different point q, the value of a_q is not equal. In the paper [ABC+15][lemma 5]'s assumption, it sets all $\alpha_1 = \alpha_q = ...$ the α (s) within a cluster to be the same.)

$$\sum_{q \in A_1} (\alpha_q - d(s, q))_+ + ... + \sum_{q \in A_1} (\alpha_q - d(s, q))_+$$

Now, we can formally define our dual solutions as,

α_q = distance from q to the nearest cluster center.

$\beta_{pq} = 0$, if q is not assigned to cluster with p as a center.

$\beta_{pq} = 0$, if q is assigned to center p.

$\beta_{pq} \geq (\alpha_q - d(p, q))_+$ if p is not a center.

$\xi = \max_p \sum_q (\alpha_q - d(p, q))_+$

And we want to examine the z^*_{dual}, the dual optimal objective value under this setting.

Since by **weak duality**, we always have

$$z^*_{dual} \leq z^*_{primal}$$

, and if we can prove that $z^*_{dual} \geq z^*_{primal}$, then by strong duality, the dual and primal optimal value equal, then the integral solution clustered by primal (i.e.) the c_i, c_j and corresponding partition A_1, A_2 should be optimal.

Proving Goal:

$$z^*_{primal} \leq (??) \sum_q \alpha_q - k \max_p \sum_q (\alpha_q - d(p,q))_+$$

Since by the assumption of the optimal solution structure we've made before, the optimal primal solution is to pick the median point of the data points, and then cut it into half from the median point. The two clusters got from this would be the optimal solution.

Then, c_1, c_2 are the two cluster centers, the primal optimal can be expressed as the following,

$$z^*_{primal} = \sum_{i=1}^{n/2} d(x_i, c_1) + \sum_{j=\frac{n}{2}+1}^{n} d(x_j, c_2)$$

Observe the expression above, this is exactly by our definition, the sum of all the α_q(s).

Then if we want to prove the goal, we need only to prove that the maximum :$\max_p \sum_q (\alpha_q - d(p,q))_+$ is zero.

8.2 1-D Proof of Integrality based on Elementary Operations on LP and comparison between fractional/integral solutions' objective value

Suppose we simulate $n/2$ points respectively from the overlapped intervals, where the intervals : 1. centers are symmetric to the origin 2. uni-radius 3. all y_p,z_{pq} satisfy the primal constraints. and we want to cluster the points into $k = 2$ clusters.

If we suppose that the optimal solution is **integral**,

then

$$y_p = [0..0...0...1...1...0..0..0]' \in \mathbb{R}^{n \times 1}$$

and

$$z_{pq} = [z_{11} z_{12} ... z_{nn}] \in \mathbb{R}^{n^2 \times 1}$$

and

$$d_{pq} = [d_{11} d_{12} ... d_{nn}] \in \mathbb{R}^{n^2 \times 1}$$

and if we suppose that two centers are i and j points, and then the corresponding z_pq vector, there will possibly be 1 among

$$[z_{i1}, z_{i2}, ..., z_{in}]$$

and

$$[z_{j1}, z_{j2}, ..., z_{jn}]$$

According to the first constraint, we can also get a series of constraints as,

$$\begin{aligned} z_{i1}+ \quad z_{j1} &= 1 \\ z_{i2}+ \quad z_{j2} &= 1 \\ &... \\ z_{in}+ \quad z_{jn} &= 1 \end{aligned}$$

And for the d_{pq}, we only care about some of the distance entries which are not zero.

Therefore, now consider the objective value:

$$d_{11}z_{11} + ... + d_{nn}z_{nn}$$

A lot of the terms above, are zero.

Only for the corresponding z_{pq} entries are not zero, we care about their distance contribution to the obj. value. i.e. the candidates are $z_{i1}, z_{i2}, ..., z_{j1}, ..., z_{jn}$

Among them, consider the very first one:

$$z_{i1}d_{i1} + z_{j1}d_{j1}(*)$$

Since for z_{i1}, z_{j1}, one of them is 0, and the other is 1, therefore, when the solution is integral, the sum of (*) should be d_{i1}/d_{j1}.

Leave the result for a while here, and consider about the non-integral scenario, if we have some fractional solutions, satisfying the primal constraints, will it be possible to get an even smaller objective value?

If we suppose that the optimal solution is **fractional**,

then

$$y_p = [0..0...0...\alpha_i...\alpha_j...1...0..0..0]' \in \mathbb{R}^{n\times 1}$$

, where $\alpha_i, \alpha_j \in (0, 1)$ and

$$z_{pq} = [z_{11}z_{12}...z_{nn}] \in \mathbb{R}^{n^2\times 1}$$

and

$$d_{pq} = [d_{11}d_{12}...d_{nn}] \in \mathbb{R}^{n^2\times 1}$$

Similarly, the obj.val

$$d_{11}z_{11} + ... + d_{nn}z_{nn}$$

will have a lot of zero terms.

And we try play with the first one,

$$\alpha d_{i1} + (1-\alpha)d_{j1}$$

Consider the geometric feature,$d_{i1} + (1-\alpha)d_{j1}$ is smalled than d_{j1}, , but bigger than our d_{i1}.

If we sum all of them up, does it mean the most optimal of fractional is sightly larger then the smallest integral solution?

8.3 Total Unimodularity

9 Week Nine (Aug 20)

9.1 Higher-dimension Proof Idea

– Idea of Dual Approach

Recall that the Primal LP and the Dual LP of our interest:

Primal LP:

$$\min_{z \in \mathbb{R}^{n \times n}} \sum_{p,q \in P} d(p,q) z_{pq}$$

$$s.t. \sum_{p \in P} z_{pq} = 1, \forall q \in P$$

$$z_{pq} \leq y_p, \forall p, q \in P$$

$$\sum_{p \in P} y_p = k$$

$$z_{pq}, y_p \in [0,1]$$

Dual LP:

$$\max_{\alpha \in \mathbb{R}^n} \sum_{q \in P} \alpha_q - k\xi$$

$$s.t. \alpha_q \leq \beta_{pq} + d(p,q), \forall p, q \in P$$

$$\sum_{q} \beta_{pq} \leq \xi, \forall p \in P$$

$$\beta_{pq} \geq 0, \forall p, q \in P$$

Recall the Fact: (Weak Duality)

For any feasible Primal/Dual solution, the primal objective is greater or equal than the dual objective,

$$Primal\ Obj \geq Dual\ Obj$$

(Here in our study case, the primal feasible solution variable is y, z and the dual feasible solution variable is α, β, ξ.) *Observe:*

If we can find feasible dual solutions s.t.

$$\text{Primal Obj.} = \text{Dual Obj.}(*)$$

then the primal solution y, z is optimal.

Strategy to prove optimality of primal solution: - To produce the dual solution that satisfies (*).

- In the approach given by the prior work [ABC+15], the dual variable α represents the **distance thresholds**.
- Intuitively, a point in the set A_j can only "see" other points within a distance α_j.

- When the input data has clear clustering structure.(i.e. the input data set satisfies the condition of *separation,center dominance* defined in [ABC+15]), the *dual certificate* α can be easily found as a distance slightly larger than the cluster centers' distance.

Directly applying to the input data has no clustering feature will lead to failure because with different input data feature, the separation can vary. The value of α_q for each cluster may vary.

- **Dual Certificate we've tested**:
  - set the dual certificate variable α - equal in each "cluster-like group"
  - set the dual certificate variable α - not equal in each "cluster-like group"
  - all α value experimented is less than 1, i.e. $\alpha < 1$ and $d < 2$. (Based on the experiment of disjoint uni-circles, constraint of the value of dual certificate is $\alpha > 1$, the distance between cluster centers is $d > 2$).
- Test results for 1D

9.2 Future Work

For the conducting future work, we can keep on discovering the following three aspects:

- Geometric meaning of dual certificate α

  Find the new definition of the α, i.e. discovering the new possible distance α represented, in terms of different input data sets.

- $\mathbb{R}^1, k > 2$

  Continue experimenting on cases when clustering factor is greater than 2, for 1-D. In other words, current project examples only include clustering the input data sets into two clusters. In the future, we can keep doing experiments for more than two clusters.

- Overlapping circles on $\mathbb{R}^d$

  The result we have only contains two overlapping circles on $\mathbb{R}^2$, due to the restriction of MATLAB simulation toolbox, we can only draw pictures on $\mathbb{R}^2$. Without the illustration of clustering result pictures, we can discover new tools and illustration ways to represent clustering results for higher dimensions. And then continue discovering the performance of the k-median clustering algorithm.

- Real life data input

  Experiment on real data set from kaggle.

References

[ABC+15] Pranjal Awasthi, Afonso S. Bandeira, Moses Charikar, Ravishankar Krishnaswamy, Soledad Villar, and Rachel Ward. Relax, no need to round: integrality of clustering formulations. In *ITCS'15—Proceedings of the 6th Innovations in Theoretical Computer Science*, pages 191–200. ACM, New York, 2015.