
Assignment 04 - 21307140003(Qirun Dai)

Table of Contents

Problem 01	1
Problem 02	2
Problem 03	3
Problem 04	4
Problem 05	6
Problem 06	7

Problem 01

```
cvx_begin
    variable x(3);
    obj = (x(1)+x(2))^2 + x(2)^2 + x(3)^2 + 3*x(1) - 4*x(2);
    a = [sqrt(2)*(x(1)+0.25*x(2)), sqrt(31/8)*x(2), 2];

    minimize( obj );
    subject to
        norm( a ) + quad_over_lin( x(1)-x(2)+x(3)+1, x(1)+x(2) ) <= 6;
        x >= 1;
cvx_end

fprintf('Optimal solution is:')
x
```

```
Calling SeDuMi 1.3.4: 20 variables, 8 equality constraints
    For improved efficiency, SeDuMi is solving the dual problem.
-----
SeDuMi 1.3.4 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.
Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta =
0.500
eqs m = 8, order n = 15, dim = 21, blocks = 6
nnz(A) = 28 + 0, nnz(ADA) = 34, nnz(L) = 21
it :      b*y      gap    delta  rate   t/tP*   t/tD*   feas cg cg   prec
0 :          1.62E+01 0.000
1 :   3.11E+00 5.39E+00 0.000 0.3322 0.9000 0.9000   1.97  1  1  1.8E+00
2 :  -1.31E+00 2.00E+00 0.000 0.3714 0.9000 0.9000   0.83  1  1  1.7E+00
3 :  -1.99E+00 4.00E-01 0.000 0.1996 0.9000 0.9000   1.17  1  1  4.4E-01
4 :  -1.94E+00 8.76E-02 0.000 0.2190 0.9000 0.9000   1.30  1  1  7.9E-02
5 :  -2.00E+00 3.34E-03 0.000 0.0382 0.9900 0.9900   1.09  1  1  2.8E-03
6 :  -2.00E+00 2.50E-07 0.314 0.0001 1.0000 1.0000   1.00  1  1  2.1E-07
7 :  -2.00E+00 8.94E-09 0.496 0.0358 0.9904 0.9900   1.00  3  3  7.7E-09

iter seconds digits      c*x      b*y
```

```
7      0.0    8.0 -1.9999999750e+00 -1.9999999941e+00
|Ax-b| = 2.4e-09, [Ay-c]_+ = 3.6E-09, |x|= 8.6e+00, |y|= 5.3e+00
```

Detailed timing (sec)

```
      Pre      IPM      Post
7.001E-03    2.900E-02    2.002E-03
Max-norms: ||b||=3, ||c|| = 6,
Cholesky |add|=1, |skip| = 0, ||L.L|| = 4.00025.
```

Status: Solved

Optimal value (cvx_optval): +5

Optimal solution is:

x =

```
1.0000
1.0000
1.0000
```

Problem 02

```
cvx_begin
    variable x(4);
    minimize ( x(1)+x(2)+x(3)+x(4) );
    subject to
        (x(1) - x(2))^2 + (x(3)+2*x(4))^4 <= 5;
        x(1) + 2*x(2) + 3*x(3) + 4*x(4) <= 6;
        x >= 0;
cvx_end
```

```
fprintf('Optimal solution is:')
x
```

Calling SeDuMi 1.3.4: 15 variables, 8 equality constraints

SeDuMi 1.3.4 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500

eqs m = 8, order n = 13, dim = 16, blocks = 4

nnz(A) = 27 + 0, nnz(ADA) = 44, nnz(L) = 26

it :	b*y	gap	delta	rate	t/tP*	t/tD*	feas	cg	cg	prec
0 :		9.02E+00	0.000							
1 :	-1.02E+00	3.03E+00	0.000	0.3363	0.9000	0.9000	2.71	1	1	2.2E+00
2 :	-7.10E-01	1.02E+00	0.000	0.3376	0.9000	0.9000	1.82	1	1	1.1E+00
3 :	-1.73E-01	2.10E-01	0.000	0.2047	0.9000	0.9000	1.30	1	1	4.8E-01
4 :	-4.19E-03	6.16E-03	0.000	0.0294	0.9900	0.9900	1.05	1	1	4.7E-01
5 :	-1.13E-08	1.43E-08	0.000	0.0000	1.0000	1.0000	1.00	1	1	4.1E-06
6 :	8.28E-15	4.71E-15	0.000	0.0000	1.0000	1.0000	1.00	1	1	9.3E-13

iter	seconds	digits	c*x	b*y
6	0.0	Inf	1.8159556866e-16	8.2757120847e-15

$|Ax-b| = 1.2e-14$, $[Ay-c]_+ = 1.2E-15$, $|x| = 6.6e+00$, $|y| = 1.4e+00$

Detailed timing (sec)

Pre	IPM	Post
6.991E-03	2.100E-02	1.006E-03

Max-norms: $\|b\|=6$, $\|c\|=2$,
Cholesky $|add|=0$, $|skip|=0$, $\|L.L\|=1$.

Status: Solved

Optimal value (cvx_optval): +1.81596e-16

Optimal solution is:

x =

1.0e-14 *

0.0406
0.0648
0.0662
-0.1534

Problem 03

```
cvx_begin
    variable x(3)
    obj1 = abs(2*x(1) + 3*x(2) + x(3));
    obj2 = square_pos(norm(x));
    obj3 = norm([sqrt(2)*(x(1)+x(2)), sqrt(5)*(x(2)+1), 1]);

    minimize (obj1 + obj2 + obj3)
    subject to
        quad_over_lin([x(1), 1], x(2)) + 2*(x(1)+x(2)+0.5*x(3))^2 + 3*x(2)^2 +
        19/2*x(3)^2 <= 7;
        max([x(1)+x(2), x(3), x(1)-x(3)]) <= 19;
        x(1) >= 0;
        x(2) >= 1;
cvx_end

fprintf('Optimal solution is:')
x
```

Calling SeDuMi 1.3.4: 35 variables, 14 equality constraints
For improved efficiency, SeDuMi is solving the dual problem.

SeDuMi 1.3.4 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500

eqs m = 14, order n = 26, dim = 36, blocks = 8

nnz(A) = 58 + 0, nnz(ADA) = 98, nnz(L) = 56

it	b*y	gap	delta	rate	t/tP*	t/tD*	feas	cg	cg	prec
0	:	3.35E+01	0.000							

```

1 : -2.45E-02 1.24E+01 0.000 0.3694 0.9000 0.9000 2.89 1 1 3.0E+00
2 : -1.42E+00 5.03E+00 0.000 0.4066 0.9000 0.9000 1.60 1 1 1.3E+00
3 : -4.41E+00 2.11E+00 0.000 0.4202 0.9000 0.9000 0.85 1 1 8.0E-01
4 : -6.95E+00 6.15E-01 0.000 0.2908 0.9000 0.9000 0.64 1 1 2.4E-01
5 : -8.03E+00 1.64E-01 0.000 0.2669 0.9000 0.9000 0.83 1 1 6.8E-02
6 : -8.45E+00 3.34E-02 0.000 0.2037 0.9000 0.9000 1.02 1 1 1.4E-02
7 : -8.53E+00 8.13E-03 0.000 0.2432 0.9008 0.9000 1.06 1 1 3.3E-03
8 : -8.53E+00 1.21E-05 0.133 0.0015 0.9000 0.0000 1.03 1 1 9.0E-04
9 : -8.54E+00 1.39E-06 0.000 0.1146 0.9153 0.9000 1.01 1 1 1.4E-04
10 : -8.55E+00 1.13E-07 0.000 0.0814 0.9900 0.9902 1.02 1 1 1.2E-05
11 : -8.55E+00 8.99E-09 0.124 0.0794 0.9900 0.9548 1.00 2 3 9.3E-07
12 : -8.55E+00 1.55E-09 0.175 0.1730 0.9104 0.9000 1.00 1 3 1.5E-07
13 : -8.55E+00 4.54E-10 0.130 0.2919 0.9240 0.9000 1.00 3 3 4.1E-08
14 : -8.55E+00 2.11E-11 0.000 0.0465 0.7911 0.9900 1.00 3 3 2.4E-09

```

```

iter seconds digits      c*x          b*y
14      0.0    Inf -8.5505013425e+00 -8.5505013094e+00
|Ax-b| = 6.1e-10, [Ay-c]_+ = 1.3E-08, |x|= 9.5e+00, |y|= 8.6e+00

```

Detailed timing (sec)

```

      Pre      IPM      Post
7.001E-03    3.100E-02    9.958E-04
Max-norms: ||b||=1, ||c|| = 19,
Cholesky |add|=1, |skip| = 0, ||L.L|| = 3.
-----

```

Status: Solved

Optimal value (cvx_optval): +8.5505

Optimal solution is:

```

x =

-0.0000
1.0000
-0.4317

```

Problem 04

```

cvx_begin
    variable x(3)
    obj1 = norm([sqrt(2)*(x(1)+x(2)), x(2), x(3), sqrt(7)]);
    obj2 = square_pos(x(1)^2 + x(2)^2 + x(3)^2 + 1);

    minimize (obj1 + obj2);
    subject to
        quad_over_lin(x(1)+x(2), x(3)+1) + x(1)^8 <= 7;
        (x(1) + x(2) + x(3))^2 + 3*x(3)^2 <= 10;
        square_pos(abs(x(1)+x(2)-x(3))) <= 20;
        x >= 0;
cvx_end

fprintf('Optimal solution is:')
x

```

Calling SeDuMi 1.3.4: 50 variables, 20 equality constraints
For improved efficiency, SeDuMi is solving the dual problem.

SeDuMi 1.3.4 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500

eqs m = 20, order n = 37, dim = 51, blocks = 13

nnz(A) = 77 + 0, nnz(ADA) = 128, nnz(L) = 76

it :	b*y	gap	delta	rate	t/tP*	t/tD*	feas	cg	cg	prec
0 :		2.11E+01	0.000							
1 :	2.92E+00	9.20E+00	0.000	0.4358	0.9000	0.9000	3.26	1	1	2.0E+00
2 :	-1.09E+00	4.60E+00	0.000	0.5000	0.9000	0.9000	2.30	1	1	8.6E-01
3 :	-2.46E+00	1.14E+00	0.000	0.2477	0.9000	0.9000	1.73	1	1	2.3E-01
4 :	-2.89E+00	5.58E-01	0.000	0.4899	0.9000	0.9000	0.82	1	1	1.2E-01
5 :	-3.32E+00	2.05E-01	0.000	0.3663	0.9000	0.9000	0.89	1	1	4.6E-02
6 :	-3.51E+00	7.47E-02	0.000	0.3653	0.9000	0.9000	0.89	1	1	1.8E-02
7 :	-3.64E+00	5.25E-03	0.000	0.0702	0.9900	0.9900	1.01	1	1	1.3E-03
8 :	-3.64E+00	7.16E-06	0.000	0.0014	0.9000	0.0000	1.01	1	1	3.5E-04
9 :	-3.64E+00	1.58E-06	0.000	0.2213	0.9105	0.9000	1.01	1	1	7.6E-05
10 :	-3.64E+00	4.12E-07	0.000	0.2598	0.9056	0.9000	1.01	1	1	1.9E-05
11 :	-3.65E+00	1.26E-07	0.000	0.3048	0.9035	0.9000	1.01	1	1	5.6E-06
12 :	-3.65E+00	2.61E-08	0.000	0.2083	0.9000	0.9010	1.00	1	1	1.4E-06
13 :	-3.65E+00	8.91E-09	0.000	0.3409	0.9000	0.9000	1.00	1	1	4.7E-07
14 :	-3.65E+00	2.52E-09	0.000	0.2829	0.9057	0.9000	1.00	1	1	1.3E-07
15 :	-3.65E+00	6.84E-10	0.000	0.2715	0.9000	0.9016	1.00	1	1	3.5E-08
16 :	-3.65E+00	1.79E-10	0.000	0.2615	0.9000	0.8956	1.00	1	1	9.5E-09

iter	seconds	digits	c*x	b*y
16	0.0	Inf	-3.6457509415e+00	-3.6457509267e+00

|Ax-b| = 7.4e-09, [Ay-c]_+ = 4.4E-08, |x|= 4.2e+00, |y|= 9.2e+00

Detailed timing (sec)

Pre	IPM	Post
7.996E-03	2.701E-02	9.958E-04

Max-norms: ||b||=1, ||c|| = 2.100000e+01,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 2297.59.

Status: Solved

Optimal value (cvx_optval): +3.64575

Optimal solution is:

x =

1.0e-04 *

0.8288
0.7660
0.8844

Problem 05

```
cvx_begin
    variable x(3)
    obj0 = square_pos(quad_over_lin(x(1), x(2)) + quad_over_lin(x(2), x(1)));
    obj1 = abs(x(1)+5);
    obj2 = abs(x(2)+5);
    obj3 = abs(x(3)+5);

    minimize( obj0 + obj1 + obj2 + obj3 )
    subject to
        square_pos(square_pos(x(1)^2 + x(2)^2 + x(3)^2 + 1) + 1) + x(1)^4 +
        x(2)^4 + x(3)^4 <= 200;
        max([(x(1)+2*x(2))^2 + 5*x(2)^2, x(1), x(2)]) <= 40;
        x(1) >= 1;
        x(2) >= 1;
cvx_end

fprintf('Optimal solution is:')
x
```

Calling SeDuMi 1.3.4: 67 variables, 29 equality constraints

For improved efficiency, SeDuMi is solving the dual problem.

SeDuMi 1.3.4 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500

eqs m = 29, order n = 52, dim = 68, blocks = 17

nnz(A) = 108 + 0, nnz(ADA) = 163, nnz(L) = 108

it	b*y	gap	delta	rate	t/tP*	t/tD*	feas	cg	cg	prec
0		7.88E+01	0.000							
1	3.79E+01	2.74E+01	0.000	0.3480	0.9000	0.9000	3.83	1	1	1.9E+00
2	4.52E+00	8.91E+00	0.000	0.3249	0.9000	0.9000	1.73	1	1	6.1E-01
3	-8.92E+00	2.99E+00	0.000	0.3355	0.9000	0.9000	1.51	1	1	2.5E-01
4	-1.37E+01	9.85E-01	0.000	0.3295	0.9000	0.9000	1.20	1	1	6.8E-02
5	-1.52E+01	4.53E-01	0.000	0.4594	0.9000	0.9000	0.63	1	1	4.1E-02
6	-1.64E+01	2.67E-01	0.000	0.5894	0.9000	0.9000	0.36	1	1	3.0E-02
7	-1.73E+01	1.58E-01	0.000	0.5932	0.9000	0.9000	0.28	1	1	2.4E-02
8	-1.90E+01	5.35E-02	0.000	0.3378	0.9000	0.9000	0.66	1	1	9.8E-03
9	-1.97E+01	2.17E-02	0.000	0.4062	0.9000	0.9000	0.63	1	1	5.1E-03
10	-2.01E+01	4.97E-03	0.000	0.2290	0.9000	0.9000	0.96	1	1	1.2E-03
11	-2.02E+01	2.14E-04	0.000	0.0430	0.9900	0.9900	1.00	1	1	6.3E-05
12	-2.02E+01	8.71E-09	0.334	0.0000	0.9000	0.0000	1.00	1	1	1.1E-05
13	-2.02E+01	2.75E-10	0.000	0.0316	0.9908	0.9900	1.00	1	1	5.4E-07
14	-2.02E+01	5.02E-11	0.000	0.1827	0.4775	0.9000	1.00	3	3	1.2E-07
15	-2.02E+01	1.33E-11	0.000	0.2656	0.7800	0.9000	1.00	3	3	3.3E-08
16	-2.02E+01	1.17E-12	0.000	0.0879	0.9900	0.9900	1.00	3	3	2.9E-09

iter	seconds	digits	c*x	b*y
16	0.0	Inf	-2.0216691566e+01	-2.0216691426e+01

|Ax-b| = 8.6e-10, [Ay-c]_+ = 1.5E-07, |x| = 1.4e+01, |y| = 2.0e+01

Detailed timing (sec)

Pre	IPM	Post
7.001E-03	3.300E-02	2.002E-03

Max-norms: $\|b\|=1$, $\|c\| = 2.010000e+02$,
Cholesky $|add|=1$, $|skip| = 0$, $\|L.L\| = 61.4277$.

Status: Solved

Optimal value (cvx_optval): +20.2167

Optimal solution is:

x =

1.0000
1.0000
-0.7833

Problem 06

```
rand('seed',21307140003);
x=rand(40,1);
y=rand(40,1);
class=[2*x<y+0.5]+1;
A1=[x(find(class==1)),y(find(class==1))];
A2=[x(find(class==2)),y(find(class==2))];
plot(A1(:,1),A1(:,2),'*','MarkerSize',6)
hold on
plot(A2(:,1),A2(:,2),'d','MarkerSize',6)
hold on

x = [A1; A2]; % The set of all points.
b = [ones(21,1); -1*ones(19,1)]; % The set of corresponding labels.
cvx_begin
    variable w(3);
    minimize ( 0.5*(w(1)^2 + w(2)^2) );
    subject to
        for i=1:40
            b(i)*([w(1), w(2)]*x(i,:) + w(3)) >= 1;
        end
cvx_end

fplot(@(x) -1/w(2)*(w(1)*x+w(3)), [0,1])
hold off
fprintf('The maximum-margin line is: %fx+%fy+%f = 0\n', w(1), w(2), w(3));

fprintf('Optimal solution is:')
w
```

Calling SeDuMi 1.3.4: 46 variables, 5 equality constraints
For improved efficiency, SeDuMi is solving the dual problem.

SeDuMi 1.3.4 by AdvOL, 2005-2008 and Jos F. Sturm, 1998-2003.

Alg = 2: xz-corrector, Adaptive Step-Differentiation, theta = 0.250, beta = 0.500

eqs m = 5, order n = 45, dim = 47, blocks = 3

nnz(A) = 126 + 0, nnz(ADA) = 15, nnz(L) = 10

it	b*y	gap	delta	rate	t/tP*	t/tD*	feas	cg	cg	prec
0		4.23E+00	0.000							
1	-5.63E+00	1.87E+00	0.000	0.4422	0.9000	0.9000	-3.36	1	1	4.4E+01
2	-1.95E+01	7.10E-01	0.000	0.3793	0.9000	0.9000	-0.95	1	1	1.6E+01
3	-4.86E+01	2.45E-01	0.000	0.3453	0.9000	0.9000	0.75	1	1	7.8E+00
4	-1.02E+02	9.34E-02	0.000	0.3809	0.9000	0.9000	-0.15	1	1	5.8E+00
5	-1.85E+02	4.04E-02	0.000	0.4322	0.9000	0.9000	-0.25	1	1	4.4E+00
6	-3.51E+02	1.19E-02	0.000	0.2958	0.9000	0.9000	0.00	1	1	2.0E+00
7	-4.62E+02	3.66E-03	0.000	0.3062	0.9000	0.9000	0.36	1	1	8.1E-01
8	-5.47E+02	1.15E-04	0.000	0.0315	0.9900	0.9900	0.78	1	1	2.8E-02
9	-5.49E+02	3.16E-06	0.000	0.0274	0.9900	0.9900	1.01	1	1	7.6E-04
10	-5.49E+02	1.54E-07	0.000	0.0487	0.9903	0.9900	1.01	1	1	3.8E-05
11	-5.49E+02	4.16E-08	0.000	0.2707	0.9000	0.9052	1.00	1	1	1.0E-05
12	-5.49E+02	5.96E-10	0.000	0.0143	0.9902	0.9900	1.00	1	1	1.5E-07
13	-5.49E+02	1.00E-10	0.000	0.1679	0.9060	0.9000	1.00	1	1	2.5E-08
14	-5.49E+02	6.31E-12	0.238	0.0631	0.9901	0.9900	1.00	1	1	1.5E-09

iter	seconds	digits	c*x	b*y
14	0.0	Inf	-5.4904627374e+02	-5.4904627334e+02

|Ax-b| = 7.8e-10, [Ay-c]_+ = 5.4E-10, |x|= 7.4e+02, |y|= 8.7e+02

Detailed timing (sec)

Pre	IPM	Post
7.001E-03	2.300E-02	2.002E-03

Max-norms: ||b||=5.000000e-01, ||c|| = 1.000000e+00,
Cholesky |add|=0, |skip| = 0, ||L.L|| = 57.6688.

Status: Solved

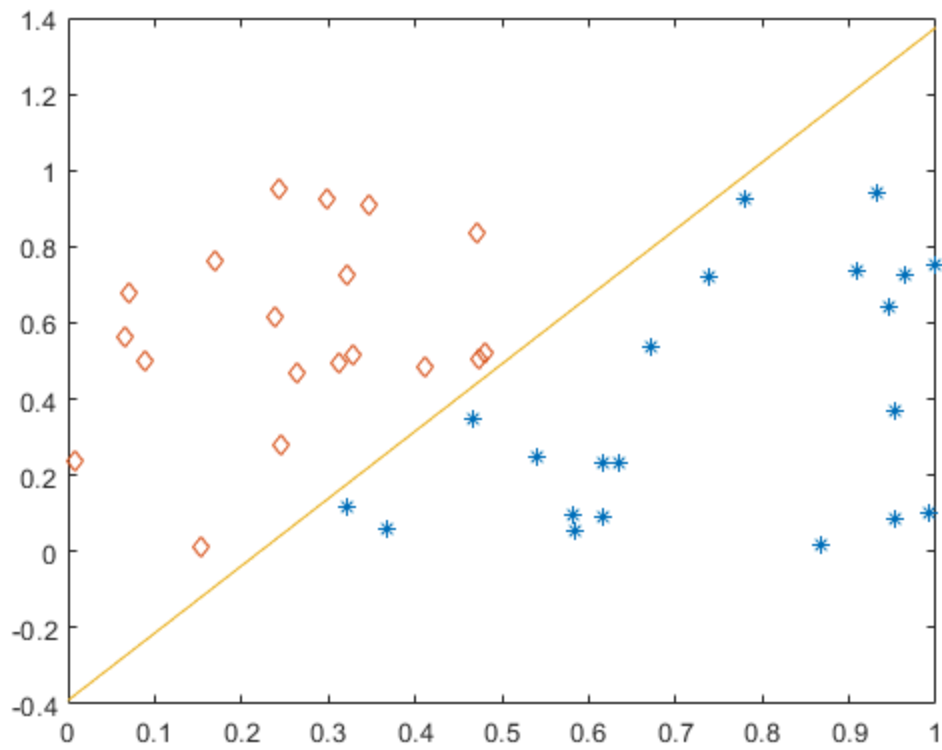
Optimal value (cvx_optval): +549.046

The maximum-margin line is: 28.833995x+-16.330747y+-6.376794 = 0

Optimal solution is:

w =

28.8340
-16.3307
-6.3768



Published with MATLAB® R2021b