```
clear all;
close all;
% plotting of ellipses (constraints)
fi = linspace(0, 2*pi, 100);
x_ball = [cos(fi); sin(fi)];    % unit ball
th = -pi/3; R60n = [cos(th) sin(th); -sin(th) cos(th)];  % pos.
 rotation
th = -pi/4; R45n = [cos(th) sin(th); -sin(th) cos(th)]; % neg.
 rotation
lam1 = [2; 1]; lam2 = [1 3];       % semiaxis length
xc1 = [0; 2]; xc2 = [-1; 2];     % ellipse centre

x_ball_r1 = R60n*diag(lam1)*x_ball + xc1*ones(1,size(x_ball,2));
x_ball_r2 = R45n*diag(lam2)*x_ball + xc2*ones(1,size(x_ball,2));

invP1 = inv(R60n*diag(lam1)*diag(lam1)*R60n');  % ellipse 1
invP2 = inv(R45n*diag(lam2)*diag(lam2)*R45n');  % ellipse 2

% second order penalty function
A = [2 1; 1 3]; f = [1; -1]; c = 2;
[XX, YY] = meshgrid(-2:0.1:2, -1:0.1:3);
ZZ = zeros(size(XX));

ZZ1 = zeros(size(XX));
for i = 1:size(XX,1),
    for j = 1:size(XX,2),
        z = [XX(i,j); YY(i,j)];
        if (((quad_form(z - xc1, invP1) <= 1) && (quad_form(z - xc2,
 invP2) <= 1))),
                ZZ1(i,j) = quad_form(z, A) + f'*z + c;
        else
                ZZ1(i,j) = nan;
        end;
    end;
end;
figure(1), contour(XX, YY, ZZ1), grid on;

%figure(4), contour(XX, YY, ZZ), grid on;
hold on
    plot(x_ball_r1(1,:), x_ball_r1(2,:), 'm--',...
         x_ball_r2(1,:), x_ball_r2(2,:), 'm--');
hold off
% cvx minimization QPQC
cvx_begin
    variable x(2);
    minimize (quad_form(x, A) + f'*x + c);
    subject to
        quad_form(x - xc1, invP1) <= 1;
        quad_form(x - xc2, invP2) <= 1;
cvx_end
min_val = cvx_optval;
x_min = x;
```

```
hold on
    plot(x_min(1),x_min(2),'ks');
hold off
disp(['minimization = ', num2str(min_val),...
    ' for ', num2str(x_min(1)), '  ', num2str(x_min(2))]);
```

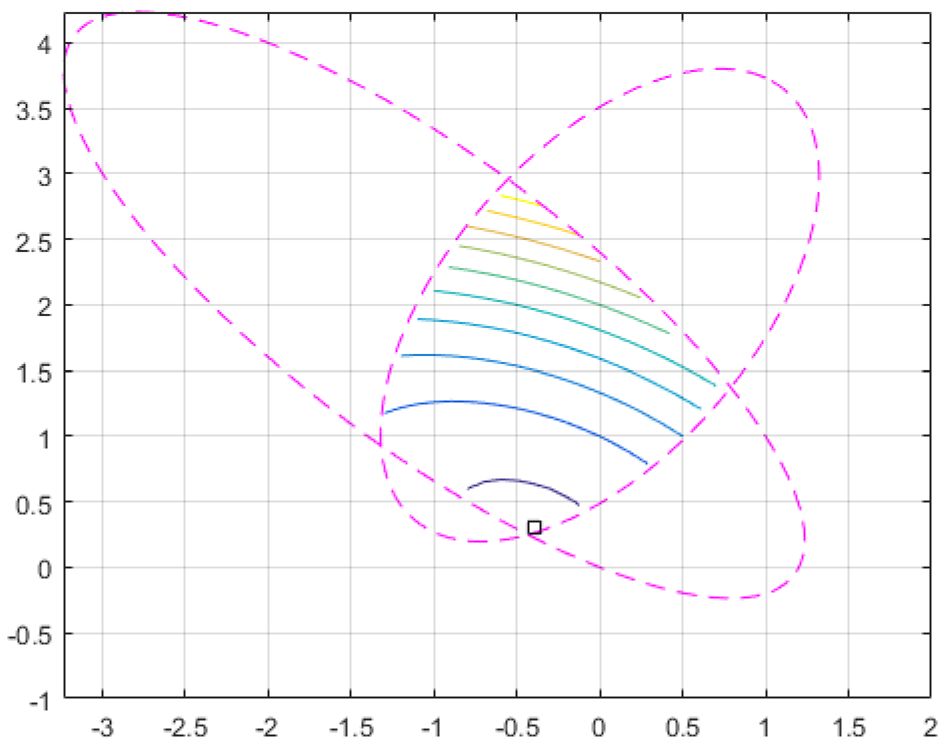
```
Calling SDPT3 4.0: 14 variables, 5 equality constraints
   For improved efficiency, SDPT3 is solving the dual problem.
------------------------------------------------------------

 num. of constraints =  5
 dim. of socp   var  = 12,   num. of socp blk  =  3
 dim. of linear var  =  2
*******************************************************************
   SDPT3: Infeasible path-following algorithms
*******************************************************************
 version  predcorr  gam  expon  scale_data
    NT      1      0.000   1        0
it pstep dstep pinfeas dinfeas  gap      prim-obj      dual-obj
 cputime
-------------------------------------------------------------------
 0|0.000|0.000|1.3e+00|3.6e+00|2.6e+02| 1.342052e+01  0.000000e+00|
 0:0:00| chol  1  1
 1|1.000|0.944|3.7e-07|2.5e-01|4.1e+01| 1.817812e+01 -6.081448e+00|
 0:0:00| chol  1  1
 2|0.947|0.995|2.4e-07|6.4e-03|4.5e+00| 2.150326e-01 -4.098536e+00|
 0:0:00| chol  1  1
 3|0.812|1.000|8.3e-07|5.2e-04|1.3e+00|-1.940218e+00 -3.255078e+00|
 0:0:00| chol  1  1
 4|1.000|0.851|8.3e-08|1.2e-04|2.7e-01|-2.737484e+00 -3.004194e+00|
 0:0:00| chol  1  1
 5|0.827|1.000|1.8e-08|5.2e-06|6.9e-02|-2.901232e+00 -2.969990e+00|
 0:0:00| chol  1  1
 6|1.000|0.810|3.3e-09|1.4e-06|1.5e-02|-2.936227e+00 -2.951314e+00|
 0:0:00| chol  1  1
 7|0.883|1.000|1.9e-09|5.3e-08|2.5e-03|-2.947239e+00 -2.949711e+00|
 0:0:00| chol  1  1
 8|1.000|0.956|1.8e-09|7.7e-09|2.7e-04|-2.949150e+00 -2.949421e+00|
 0:0:00| chol  1  1
 9|0.980|0.982|9.9e-11|4.9e-10|5.6e-06|-2.949403e+00 -2.949408e+00|
 0:0:00| chol  1  1
10|0.993|1.000|7.3e-13|2.0e-11|7.0e-08|-2.949408e+00 -2.949408e+00|
 0:0:00|
  stop: max(relative gap, infeasibilities) < 1.49e-08
-------------------------------------------------------------------
 number of iterations   = 10
 primal objective value = -2.94940800e+00
 dual   objective value = -2.94940807e+00
 gap := trace(XZ)       = 6.98e-08
 relative gap           = 1.01e-08
 actual relative gap    = 1.01e-08
 rel. primal infeas (scaled problem)   = 7.28e-13
 rel. dual     "        "       "      = 1.98e-11
```

```
 rel. primal infeas (unscaled problem) = 0.00e+00
 rel. dual     "        "       "      = 0.00e+00
 norm(X), norm(y), norm(Z) = 4.0e+00, 7.2e-01, 2.7e+00
 norm(A), norm(b), norm(C) = 4.6e+00, 6.1e+00, 4.3e+00
 Total CPU time (secs)  = 0.48
 CPU time per iteration = 0.05
 termination code       =  0
 DIMACS: 7.4e-13  0.0e+00  3.5e-11  0.0e+00  1.0e-08  1.0e-08
-------------------------------------------------------------------

------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +1.65

minimization = 1.65 for -0.4  0.3
```



Published with MATLAB® R2016b