Simple SDP Example using CVX

This script demonstrates formulating and solving SDP problems in CVX.

Contents

Requirements

This script requires the CVX toolbox: http://cvxr.com/cvx/

Problem Data in Standard Form

```
c = [1 -1];
A1 = [1 0; 0 1];
A2 = [0 1; 1 -1];
B = [1 0; 0 0];
```

Primal Problem in CVX

```
% Initialize CVX environment
cvx_begin sdp % Specify that matrix inequalities are semidefinite constraints

    % Decision variables
    variable z(2,1) % Vector decision variable

    % Objective
    minimize c*z

    % Semidefinite Constraint
    z(1)*A1 + z(2)*A2 >= B;
cvx_end
```

```
Calling SDPT3 4.0: 3 variables, 1 equality constraints
------------------------------------------------------------

 num. of constraints =  1
 dim. of sdp    var  =  2,   num. of sdp  blk  =  1
```

```
*****************************************************************
   SDPT3: Infeasible path-following algorithms
*****************************************************************
 version  predcorr  gam  expon  scale_data
   HKM      1      0.000   1        0
it pstep dstep pinfeas dinfeas  gap      prim-obj      dual-obj    cputime
-----------------------------------------------------------------
 0|0.000|0.000|5.0e-01|6.1e+00|2.0e+02| 1.000000e+01  0.000000e+00| 0:0:00| chol  1  1
 1|1.000|1.000|2.5e-07|6.4e-02|1.0e+01| 8.074566e+00 -5.519999e-01| 0:0:00| chol  1  1
 2|0.979|1.000|6.6e-08|6.4e-03|4.2e-01|-1.168985e-01 -5.240860e-01| 0:0:00| chol  1  1
 3|1.000|0.926|2.4e-08|1.1e-03|5.5e-02|-1.592323e-01 -2.125979e-01| 0:0:00| chol  1  1
 4|0.987|0.989|6.6e-09|7.4e-05|7.0e-04|-1.994838e-01 -2.000637e-01| 0:0:00| chol  1  1
 5|0.989|0.989|1.6e-10|7.1e-06|7.7e-06|-1.999943e-01 -1.999908e-01| 0:0:00| chol  1  1
 6|0.989|0.989|1.6e-11|7.8e-08|8.4e-08|-1.999999e-01 -1.999999e-01| 0:0:00| chol  1  1
 7|0.996|1.000|1.6e-13|3.2e-12|1.2e-09|-2.000000e-01 -2.000000e-01| 0:0:00|
  stop: max(relative gap, infeasibilities) < 1.49e-08
-----------------------------------------------------------------
 number of iterations   =  7
 primal objective value = -1.99999999e-01
 dual   objective value = -2.00000000e-01
 gap := trace(XZ)       = 1.16e-09
 relative gap           = 8.30e-10
 actual relative gap    = 8.26e-10
 rel. primal infeas     = 1.55e-13
 rel. dual   infeas     = 3.21e-12
 norm(X), norm(y), norm(Z) = 1.0e+00, 2.0e-01, 1.0e+00
 norm(A), norm(b), norm(C) = 2.6e+00, 2.0e+00, 2.2e+00
 Total CPU time (secs)  = 0.10
 CPU time per iteration = 0.01
 termination code       =  0
 DIMACS: 1.6e-13  0.0e+00  3.6e-12  0.0e+00  8.3e-10  8.3e-10
-----------------------------------------------------------------
-------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +0.8
```

Primal optimal value returned by CVX

```
cvx_optval
z
```

```
cvx_optval =

   8.0000e-01


z =

   1.2000e+00
   4.0000e-01
```

Dual Problem in CVX

```
% Initialize CVX environment
cvx_begin sdp % Specify that matrix inequalities are semidefinite constraints

    % Decision variables
    variable Y(2,2) % Matrix decision variable

    % Objective
    maximize trace(B*Y)

    % Linear Equality Constraints
    trace(A1*Y) == c(1);
    trace(A2*Y) == c(2);

    % Semidefinite Constraint
    Y >= 0;
cvx_end
```

```
Warning: This linear matrix inequality appears to be unsymmetric. This is
very likely an error that will produce unexpected results. Please check
the LMI; and, if necessary, re-enter the model.

Calling SDPT3 4.0: 3 variables, 1 equality constraints
   For improved efficiency, SDPT3 is solving the dual problem.
------------------------------------------------------------

 num. of constraints =  1
 dim. of sdp    var  =  2,   num. of sdp  blk  =  1
*******************************************************************
   SDPT3: Infeasible path-following algorithms
*******************************************************************
 version  predcorr  gam  expon  scale_data
   HKM      1      0.000   1        0
it pstep dstep pinfeas dinfeas  gap      prim-obj      dual-obj    cputime
-------------------------------------------------------------------
 0|0.000|0.000|5.0e-01|6.1e+00|2.0e+02| 1.000000e+01  0.000000e+00| 0:0:00| chol  1  1
 1|1.000|1.000|2.5e-07|6.4e-02|1.0e+01| 8.074566e+00 -5.519999e-01| 0:0:00| chol  1  1
 2|0.979|1.000|6.6e-08|6.4e-03|4.2e-01|-1.168985e-01 -5.240860e-01| 0:0:00| chol  1  1
 3|1.000|0.926|2.4e-08|1.1e-03|5.5e-02|-1.592323e-01 -2.125979e-01| 0:0:00| chol  1  1
 4|0.987|0.989|6.6e-09|7.4e-05|7.0e-04|-1.994838e-01 -2.000637e-01| 0:0:00| chol  1  1
 5|0.989|0.989|1.6e-10|7.1e-06|7.7e-06|-1.999943e-01 -1.999908e-01| 0:0:00| chol  1  1
 6|0.989|0.989|1.6e-11|7.8e-08|8.4e-08|-1.999999e-01 -1.999999e-01| 0:0:00| chol  1  1
 7|0.996|1.000|1.6e-13|3.2e-12|1.2e-09|-2.000000e-01 -2.000000e-01| 0:0:00|
  stop: max(relative gap, infeasibilities) < 1.49e-08
-------------------------------------------------------------------
 number of iterations   =  7
 primal objective value = -1.99999999e-01
 dual   objective value = -2.00000000e-01
 gap := trace(XZ)       = 1.16e-09
 relative gap           = 8.30e-10
 actual relative gap    = 8.26e-10
 rel. primal infeas     = 1.55e-13
 rel. dual   infeas     = 3.21e-12
 norm(X), norm(y), norm(Z) = 1.0e+00, 2.0e-01, 1.0e+00
```

```
 norm(A), norm(b), norm(C) = 2.6e+00, 2.0e+00, 2.2e+00
 Total CPU time (secs)  = 0.08
 CPU time per iteration = 0.01
 termination code       =  0
 DIMACS: 1.6e-13  0.0e+00  3.6e-12  0.0e+00  8.3e-10  8.3e-10
------------------------------------------------------------------
------------------------------------------------------------
Status: Solved
Optimal value (cvx_optval): +0.8
```

Dual optimal value returned by CVX

Note that the dual optimal value is equal to the primal optimal value

```
cvx_optval
Y
```

```
cvx_optval =

   8.0000e-01


Y =

   8.0000e-01  -4.0000e-01
  -4.0000e-01   2.0000e-01
```

CVX can also return the primal and dual optimal variables

```
% Initialize CVX environment
% The quiet key word can be used to suppress the solvers output
cvx_begin sdp quiet


    % Decision variable
    variable z(2,1)

    % Dual variable
    % The dimension of the dual variable will be inferred from the
    % corresponding constraint so it is not necesary to specify the
    % dimension
    dual variable Y

    % Objective
    minimize c*z

    % Semidefinite Constraint with corresponding dual variable Y
    Y : z(1)*A1 + z(2)*A2 >= B;
cvx_end
```

```
cvx_status
```

```
cvx_status =

Solved
```

Primal and Dual Optimal Variables

Note the primal and dual optimal variables are the same as those found previously.

```
z
Y
```

```
z =

   1.2000e+00
   4.0000e-01


Y =

   8.0000e-01  -4.0000e-01
  -4.0000e-01   2.0000e-01
```

Attribution

This example supplements the book "Networks of Dissipative Systems: Compositional Certification of Stability, Performance, and Safety" by Murat Arcak, Chris Meissen, and Andrew Packard.

Published with MATLAB® R2015a